(1)

Technical Report 765

Estimating Computer-Based Training Development Times

Joni Jay, Kenneth Bernstein, and Sonia Gunderson
Scientific Systems, Inc.

ARI Field Unit at Fort Knox, Kentucky
Training Research Laboratory

ari

U. S. Army
Research Institute for the Behavioral and Social Sciences

October 1987

U. S. ARMY RESEARCH INSTITUTE

FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency under the Jurisdiction of the

Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON
Technical Director

WM. DARRYL HENDERSON
COL, IN
Commanding

Research accomplished under contract
for the Department of the Army

Scientific Systems, Inc.

Technical review by

Scott E. Graham
Milton Koger
Theodore M. Shlechter

Accession For		
NTIS GRA&I		X
DTIC TAB		☐
Unannounced		☐
Justification		
By		
Distribution/		
Availability Codes		
Dist	Avail and/or Special	
A-1		

QUALITY INSPECTED 2

NOTICES

~~DISTRIBUTION: Primary distribution of this report has been made by ARI. Please address correspondence concerning distribution of reports to: U.S. Army Research Institute for the Behavioral and Social Sciences, ATTN: PERI-POT, 5001 Eisenhower Ave., Alexandria, Virginia 22333-5600.~~

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Technical Report 765

Estimating Computer-Based Training Development Times

Joni Jay, Kenneth Bernstein, and Sonia Gunderson
Scientific Systems, Inc.

ARI Field Unit at Fort Knox, Kentucky
Donald F. Haggard, Chief

Training Research Laboratory
Jack H. Hiller, Director

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES
5001 Eisenhower Avenue, Alexandria, Virginia 22333-5600

Office, Deputy Chief of Staff for Personnel
Department of the Army

October 1987

Army Project Number
2Q263743A794

Education and Training

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: Unclassified

1b. RESTRICTIVE MARKINGS:

2a. SECURITY CLASSIFICATION AUTHORITY:

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE:

3. DISTRIBUTION / AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S):

5. MONITORING ORGANIZATION REPORT NUMBER(S): ARI Technical Report 765

6a. NAME OF PERFORMING ORGANIZATION: Scientific Systems, Inc.

6b. OFFICE SYMBOL (If applicable):

7a. NAME OF MONITORING ORGANIZATION: U.S. Army Research Institute Fort Knox Field Unit

6c. ADDRESS (City, State, and ZIP Code): One Alewife Place, Cambridge, MA 02140

7b. ADDRESS (City, State, and ZIP Code): Fort Knox, KY 40121-5620

8a. NAME OF FUNDING / SPONSORING ORGANIZATION: U.S. Army Research Institute

8b. OFFICE SYMBOL (If applicable): PERI-IK

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: MDA903-83-C-0483

8c. ADDRESS (City, State, and ZIP Code): 5001 Eisenhower Avenue, Alexandria, VA 22333-5600

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
6.37.43.A	2Q2637-43A794	5.1.2.C.1	

11. TITLE (Include Security Classification): Estimating Computer-Based Training Development Times

12. PERSONAL AUTHOR(S): Joni Jay, Kenneth Bernstein, and Sonia Gunderson

13a. TYPE OF REPORT: Interim

13b. TIME COVERED: FROM 06/86 TO 06/87

14. DATE OF REPORT (Year, Month, Day): 1987 October 14

15. PAGE COUNT: 102

16. SUPPLEMENTARY NOTATION: Research was technically monitored by Dr. Donald F. Haggard.

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number): Computer-Based Instruction; Cost estimate; Computer-Based Training; Educational technology cost; Military training; Cost analyses; Training costs

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The Army Research Institute (ARI) and other government organizations involved in military training are major users of computer-based training (CBT). In the bidding and procurement process for CBT development contracts, there is a need for specific criteria for determining appropriate cost ranges. Part 1 provides an examination of the issues in making accurate CBT development cost estimates. Part 2 of this report provides useful descriptive information about the current methods CBT developers use to estimate costs and the range of development times required to produce CBT, as well as the factors developers believe have the greatest effect on cost. Part 3 describes current costing models and evaluates a CBT development costing tool currently on the market. The report makes recommendations for improving the CBT development costing process. The results and recommendations contained in this report could provide a basis for other researchers to develop a development costing tool to be used throughout the industry to improve the contracting process.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION: Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL: Scott E. Graham

22b. TELEPHONE (Include Area Code): (502) 624-2613

22c. OFFICE SYMBOL: PERI-IK

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

FOREWORD

The Army Research Institute (ARI) and other government organizations involved in military training are major users of computer-based training (CBT). In the bidding and procurement process for CBT development contracts, there is a need for specific criteria for determining appropriate cost ranges. Part 1 of this report provides an examination of the issues in making accurate cost estimates for CBT development. Part 2 of this report provides useful descriptive information about the current methods CBT developers use to estimate costs and the range of development items required to produce CBT, as well as the factors developers believe have the greatest effect on cost. Part 3 describes current costing models and evaluates a CBT development costing tool currently on the market.

The report also makes recommendations for improving the CBT development costing process. The results and recommendations contained in this report should provide a basis for other researchers to develop a CBT development costing tool to be used throughout the industry to improve the CBT contracting process. In October 1986, results of this research were briefed to senior representatives of the Training and Doctrine Command, the Armor School, and the Ordnance School.

Edgar M Johnson

EDGAR M. JOHNSON
Technical Director

ACKNOWLEDGMENTS

The authors gratefully acknowledge the assistance of Dr. Scott Graham, Dr. Theodore Shlechter, and Dr. Donald Haggard of the U.S. Army Research Institute and Dr. Larry Israelite, Dr. E. Webb Stacy, Ms. Ruth Ann Marco, and Mr. John Begg of Scientific Systems, Inc., for advice on the design of this study and the statistical analysis, and for comments on drafts of the report.

ESTIMATING COMPUTER-BASED TRAINING DEVELOPMENT TIMES

EXECUTIVE SUMMARY

Requirement:

The military is one of the largest users of computer-based training (CBT). CBT development project estimates have been notoriously inaccurate. The purpose of this project was to identify current costing practices, actual development times, and factors affecting CBT development times, and to examine the applicability of software cost models and validate an existing CBT development costing model.

Procedure:

In Part One, information was obtained from published and unpublished research, conversations with CBT developers, and structured interviews with project managers. In Part Two, almost 200 CBT developers were surveyed. From their descriptive data, statistical analysis was performed to determine if significant differences existed between groups of interest. In Part Three, studies on the effectiveness of software costing models were examined for their applicability to CBT. Additionally, a commercially available tool for estimating CBT development costs was validated using nine completed CBT projects.

Findings:

Fewer than 10% of CBT developers are able to estimate within 5% of actual cost. Experienced developers are no better at making estimates than inexperienced developers, but inexperienced developers are more often off by more than 20%. Developers attribute poor estimates to the following: (a) poorly defined RFPs that result in change of scope or excessive revisions; (b) lack of historical data; and (c) lack of an accurate costing method.

No standard method for measuring CBT is currently used. Although the instructional hour is the most common, it is widely disliked as inaccurate and not reflecting complexity and other factors. The lack of a standardized method makes it difficult to measure what is to be done or whether it has been done satisfactorily. Not all developers include the same tasks when costing CBT, so purchasers cannot be sure that all bids contain all steps necessary for the development of good training.

The most common method for estimating the cost of a unit of CBT is using the industry averages of 100 to 400 development hours per instructional hour. These broad ranges are virtually useless for estimating specific projects. The reported ranges of actual time required to develop a unit of CBT were from 1 to 4000 hours per hour, with 153 and 316 as the mean low and high in the survey.

A host of factors were identified as contributing to CBT development costs. The factors mentioned most often and rated as having the greatest effect on cost are these:

- Complexity of instructional design strategy;
- Clarity of project specifications at the outset and adherence to them;
- Complexity of content;
- Number of revisions;
- Complexity and number of features (e.g., graphics and help); and
- Experience of the development team as a group and individually.

One commercially available tool was found useful for novices who need a cost range estimate of CBT to determine whether to go ahead. However, like other guidelines available, the tool cannot be used with confidence to predict the cost of any particular project. Because of similarities between software and CBT development, it seems reasonable to use software metrics techniques in developing metrics for CBT.

Utilization of Findings:

The results of this research can be used for further studies to develop an accurate model for costing CBT development. Until such a model is validated, the following recommendations to make the CBT development costing process more accurate can be implemented:

- Develop more specific requests for proposals;
- Complete analysis before project bids are made;
- Consider cost factors in making estimates; and
- Emphasize student achievement rather than instructional hours.

ESTIMATING COMPUTER-BASED TRAINING DEVELOPMENT TIMES

CONTENTS

ESTIMATING COMPUTER-BASED TRAINING DEVELOPMENT TIMES

Accurate cost estimates are essential if purchasers of computer-based training (CBT) are to make informed decisions about the cost-effectiveness of CBT for particular training applications. Without clear definitions of cost variables and standardized cost models, purchasers have no realistic basis for evaluating the relative merits of CBT proposals. Clear methods for deriving cost estimates are also essential if purchasers and developers are to make compromises which can bring project development times and costs within acceptable levels.

Courseware developers commonly derive development cost estimates using general ratios of the time required to produce an hour of instruction. Because of the number of variables involved in CBT development, such general ratios have limited use for estimating costs. There is a critical need to find better methods for deriving CBT cost estimates since there presently appears to be no universally effective costing tools for CBT development. As a result, even though CBT developers make the best cost estimates they can based on past experience, the estimates are often inaccurate. Buck and Gillespie (1985) have characterized the present ability of CBT developers to predict development times: "Estimating the amount of time needed to develop courseware is so difficult you might as well assume from the outset that whatever figures you come out with will be wrong" (page 46). Often, the results are either a negotiated reduction in project scope or a request for additional funds and time to complete the project. The purposes of this research were:

1. To define the problems which contribute to the difficulty in making accurate computer-based training (CBT) development cost estimates,

2. Study current CBT development costing practices,

3. Validate a CBT costing tool, and

4. Make recommendations to improve the CBT development contract bidding and procurement process.

CBT is a complex training medium. As such, several factors cause difficulty in accurately predicting the costs of developing CBT. These include: the immaturity of the medium; number of variables

involved in making estimates; lack of developer experience; lack of standardized methods for measuring the quantity and quality of CBT; and, lack of standard data collection procedures.

Costs associated with CBT systems as a whole include lifecycle costs such as hardware, courseware, delivery, and maintenance costs. This report focuses on the costs involved in the analysis, design, and development of courseware. Research was conducted in three areas and the results are reported in separate parts :

Part 1--A review of issues in estimating CBT development costs
Part 2--A survey of CBT developers
Part 3--A review of CBT cost models and an informal validation of an existing CBT costing tool

Part One describes the issues which impact the ability of developers to make CBT cost estimates. The primary issues discussed are methods for estimating development costs, development times, and development cost factors. Sample CBT development contracts were examined for information on each of these issues.

Part Two describes the design and results of a survey conducted to gather information from almost 200 CBT developers. The survey provides information on methods CBT developers currently use to measure courseware, development time required for CBT production, factors which most impact CBT costs, and cost estimating practices.

Part Three contains a description of software metrics and how the related techniques for estimating software development costs might be applied to courseware development. Part Three also presents the results of an informal validation of an existing cost model designed to predict CBT development costs.

PART ONE: ISSUES IN ESTIMATING CBT DEVELOPMENT TIME

Method

Data for this part of the report came from three categories of sources: (a) published and unpublished studies, (b) informal conversations with professionals in CBT and related fields, and (c) structured interviews with project managers of CBT projects. The methods used to obtain information from each of these sources are described below.

Published and Unpublished Studies

A literature search was conducted to identify articles on current CBT development cost estimation practices. Libraries and on-line information services were searched. The result of the search was a custom bibliography and abstracts of selected technical reports. Articles concerning CBT development were examined to determine whether they contained information about CBT development costs.

Contacts With CBT Professionals

Because the literature on costing CBT is not extensive, a substantial effort was made to contact CBT development professionals. This was necessary to provide the broadest and most current information on CBT costing practices.

Names of CBT professionals were obtained from a variety of sources, including: (a) authors who have published in the field, (b) directories of CBT developers, (c) developers who expressed special interest in the research after participating in the survey, and (d) "networking" via professional CBT contacts. Among those contacted were military training personnel, commercial CBT developers, university researchers, and professional organizations. Appendix A contains a list of the organizations and individuals who participated in this phase of the study.

CBT Contract Information

The authors attempted to obtain quantifiable data on completed CBT projects to provide supplementary information on the CBT development process. Such data were difficult to obtain because: (a) contract reports were not readily available, (b) the government does not require systematic tracking of development times and costs, (c) most available reports contain only data that are inconsistent or irrelevant to this study, and (d) non-governmental reports are proprietary. Therefore, the authors gathered anecdotal information on a sampling of completed CBT projects.

Four developers were selected to participate in this portion of the research. Participants included the training groups of two major aerospace companies, the training division of a large computer manufacturing firm, and a military training facility. Participants were selected based on their willingness to participate without compensation, and the availability of historical data on development times of contracts completed within the last three years. Seven

contracts were examined. Of the eight, five were military, two were commercial, and one was academic.

Information was gathered by means of a structured interview. Prior to the interview, participants received a series of questions on the items listed above. Once the developers gathered the available information, telephone interviews on each contract were conducted to gather anecdotal information. All participants were assured of institutional and project anonymity and were asked to be as candid as possible.

Results/Discussion

Information gathered during the interview falls into the following categories:

- Measuring CBT,
- Estimating CBT development costs, and
- Factors affecting development times.

Methods for Measuring a Unit of CBT

The method used to measure CBT affects the ability of developers to make accurate cost estimates. CBT purchasers and developers can only make meaningful cost comparisons if they agree on a method for measuring courseware. Currently, there is no universally accepted measurement method.

The most common method for measuring CBT is counting the hours of instruction. Other methods include counting the number of screens per lesson, counting the number of interactions per hour, and counting the number of objectives or lessons in a course. Each of these methods is discussed below.

Hour of Instruction. Time and cost estimates for CBT development are most frequently made on the basis of the number of development hours required to produce the courseware for one hour of instruction. However, the definition of "one hour of CBT instruction" is not standardized. Fairweather and O'Neal (1984) characterize this method as "the most slippery metric known to man" (p. 92).

Many professionals define an hour of CBT instruction as the amount of material an average student would go through in an hour. However, a closer examination of this definition reveals several defects. In one case, the lack of a clear definition of the "average" student created negative consequences for a CBT developer. On that large CBT program, development costs were paid after the project was completed based on the amount of time it took four students to go through the instruction. In this case, "smart" students were selected and they completed the instruction very quickly, resulting in a smaller number of instructional hours than might have been expected.

The hour of instruction method may discourage the design of good courseware. One of the greatest advantages of CBT is that training can be tailored to each student's needs, so that below average students can be given considerable help through branched instruction which explains the misconceptions behind incorrect answers. If this additional instruction is not counted because the average student never sees it, developers may be discouraged from designing courseware which meets the needs of students who require it most.

Another definition which attempts to address this problem defines the hours of instruction in a program as the number of hours it takes an average student to go through every screen in the courseware, including feedback for all possible wrong answers and all help screens. This approach might be more useful because it reflects the total amount of instruction developed. It does have the disadvantage that developers might add superfluous screens to increase unnecessarily the number of hours in a course.

Screens and Interactions. Some professionals prefer to avoid the issue of time altogether when measuring courseware, choosing instead to measure the total number of screens per lesson. This method is attractive because it is simple and concrete. However, it does not take into account the complexity of the screens or the computer code. It is not valid to compare two lessons which each contain 150 screens if one consists primarily of text displays and the other utilizes help screens, questions, glossaries, and technical diagrams. The number of screens is not sufficient as a sole method of measuring CBT.

An interesting variation on this approach has been developed by R. Yeager (personal communication, January, 1987) who defines "an hour of instruction" as approximately 60 interactions. Yeager believes that simply using time as a method of measuring courseware does not reflect the quality of the courseware and discourages CBT developers from designing more interactive courseware because more interactions increase development time and cost.

Yeager's method of measurement begins to address the issue of courseware quality. He believes that for effective learning to occur, interactions must be frequent and meaningful. He defines a "meaningful interaction" as one that has three characteristics:

- It must be related to the lesson objectives.
- It must occur within a situation where there is a broad range of expected answers and the student must discriminate between several choices.
- There must be specific feedback for each answer, including unanticipated responses.

There are at least two problems with Yeager's method of measuring CBT. First, it does not give credit for "page turning" in which the student reads one screen after another. Although most CBT experts agree that "electronic page turning," often is not the most effective use of the medium, this approach is appropriate for some applications. Second, this method of measurement may encourage developers to include superfluous questions which require the student to parrot information in order to keep the number of interactions high. In any case, the effectiveness of Yeager's method of CBT measurement has yet to be measured.

Objectives and Lessons. Other developers measure CBT by the number of objectives or lessons to be taught. For example, L. Wilson (personal communication, September, 1986) of Ford Aerospace and Communication Corporation (formerly the Hazeltine Corporation Training Systems Center) reported that they base their cost estimates on the number and type of lesson segments required. Based on their experience, they estimate costs separately for lesson types that require simulations, cognitive content, remedial instruction, video, etc. This method seems to be an improvement over the arbitrary "hour," but can only be used if the objectives or lessons can be clearly defined before the contract begins.

The Army Training Support Center (ATSC) is currently conducting a study of training costs. Because of the problems already discussed regarding the hour as a method of CBT measurement, ATSC decided to use performance objectives as the unit of measure. Using performance objectives was possible because the front-end analysis was completed, the training delivery medium selected, and a sample segment of instruction identified in advance to serve as a model for creating performance objectives. A spokesperson for ATSC admitted that

performance objectives still can be "nebulous." ATSC will evaluate the data in mid-1987 to decide if performance objectives are a valid and practical method of CBT measurement.

In summary, although the hour of instruction is the most common measurement noted, various methods exist, each with its own advantages and disadvantages. None of the methods have been validated to date, and in the end most people agree that the methods used are only a "best guess".

Methods for Estimating CBT Development Costs

Request for Proposals. A CBT developer must prepare cost estimates based on the description of the statement of work contained within the Request For Proposal (RFP). The RFP is usually the major source of information available to the developer. RFPs provide developers with information such as the scope of the project, the type of instruction required, and the intended audience. A statement of work covers the entire project and bids typically must cover all tasks, from front-end analysis through evaluation.

RFPs vary in the quantity and quality of descriptive information. The CBT developer's ability to prepare complete and accurate estimates is dependent upon the level of the descriptive detail contained within the RFP. For example, one experienced developer described a typical RFP which contained a detailed description of the organization and audience, but no course objective, or discussion of desired strategies or features. Only general list of topics to be covered was provided. Several months after the contract began, the client insisted that very expensive features and strategies, including case study simulation and animation, be used. Unfortunately, the developer had bid the project assuming that the most simple types of strategies and features appropriate for the topics would be used.

Some purchasers do provide details that make estimating development costs much easier. The same developer described an RFP from a commercial organization which included detailed objectives with performance criteria, definitions and ratings of types of instructional strategies, features and graphics desired for each objective, and an explanation of exactly how the organization expected to participate in the review process. The developer reported that estimating costs was easy and expressed confidence that the estimate was relatively accurate. Of course, many purchasers are not training design experts and are not able to articulate their requirements so specifically. However, several developers have noted that preconceived ideas about the nature and scope of the content often surface long after the bid has been made.

CBT developers typically bid on contracts prior to conducting a front-end analysis. Training tasks, lessons, and instructional approaches are usually specified after costs have been estimated and the project is underway. Thus, bids are usually made with incomplete information about the size of the project. Once the developer has prepared an estimate and submitted it to the government, the contracting agency usually selects the CBT developer based on the lowest bid best value among technically qualified proposals.

Estimation Accuracy. Developers and experts alike have noted that developers currently make poor estimates for CBT development projects. Mikos, Sullivan and Casey (1987) examined the ability of experienced CBT developers to estimate costs in a study to identify cost factors. A group of experienced developers attending the 1987 National Society for Performance and Instruction conference (n=18) was asked to estimate the cost of an hour of courseware, based on specifications for a unit that had been developed by an organization known to the researchers. The actual cost of development of the unit was also known to the researchers. The CBT developers who had experience in estimating CBT costs as well as in developing CBT (n=6) estimated more accurately; 67% estimated within 25% of the actual cost. Of developers without CBT costs estimating experience (n=12), only 10% came within 25% of the actual cost. The latter group's estimates ranged from 82% under to 84% over the actual cost.

The subjects were then assisted by the researchers in developing a "project complexity multiplier" which included factors such as "client personality," politics/corporate culture, and anticipated availability of SMEs. The inexperienced work load estimators improved markedly when their multipliers were used; 45% came within 25% of actual cost. The performance of the group with previous work load estimating experience did not improve with the multiplier; again 67% were within 25% of actual costs.

The improved CBT cost estimates of the inexperienced work load estimators in this small study suggest that developers can make better estimates when project complexity factors are taken into account. Mikos, et. al., suggest that the experienced work load estimators were already accounting for the complexity factors in their original estimates.

One reported cause of actual costs exceeding estimates is that costs proposed for projects are often known to be low at the time of bidding. Several anonymous contributors to this research admitted that their organizations deliberately underbid contracts to "get the business" in this highly competitive field. One developer reported that his organization was willing to break even or take a loss to stay in the business. Another admitted that his organization bids at the

"industry average" of 200 development hours per instructional hour even though they have never produced courseware at that price. He added that all of their projects have had "special circumstances" which made them cost more than they should have. Many of these developers expressed the expectation that their next project would be free of such "special circumstances" and could be produced within their optimistic estimates. Referring to the same expectation in software developers, Brooks (1985) said it is a serious mistake to assume that "all will go well."

In order to make estimates, the developers generally determine how many units of CBT will be developed and how many hours it will take to develop each unit. The following section focuses on how development cost are estimated.

Estimation methods. Currently most developers use one of two methods to determine how much time it will take to develop a unit (most often an instructional hour) of CBT. The most common method is to use "industry averages." Dean and Whitlock (1983) note that "Many statistics have been produced for the time it takes, in person-hours, to produce one hour of a CBT course. The figures generally vary between 100:1 and 200:1." Kearsley (1983) states that the development of one hour of courseware takes from 100 to 400 hours. Orlansky and String (1979) found that authoring and coding one hour of instruction took took from 80 to 830 personhours per instructional hour.

Such ratios have limited utility for deriving cost estimates because of the number of variables involved in CBT development. While these ratios may be useful as general guidelines, many developers interviewed noted that these ratios cannot provide specific enough estimates on which to base accurate bids. Without any guidance in how to apply the ratios based on project-specific factors, developers choosing the middle of the range (say 200 hours per hour) might over or underbid by 100% or more. For example, using 100 to 400 hours per hour as the range, the developer may assume that 100 hours would be required for a simple tutorial lesson with limited graphics and features, while 400 hours may be required for a complex simulation with many graphics and multiple paths. However, if the tutorial includes remedial loops, interactive video sequences, customized feedback, and if the lessons must be written completely from scratch and reviewed by four subject-matter experts from across the country, the development time may easily exceed that of the simulation. Clearly these ratios alone cannot provide enough guidance for developers to make accurate estimates for specific projects.

Although most experts agree that a database compiled from previous CBT development efforts is essential for deriving accurate time and cost estimates, few developers interviewed are currently

using historical data effectively to make cost estimates. H.J. O'Neil (personal communication, January, 1987) reported that he is not aware of organizations which have collected reliable data on CBT development. He said most CBT cost estimates are not derived systematically; they are based on educated guesses or intuition. G. Gery (personal communication, February, 1987) said that, although some developers are tracking development times, she regards most data gathered and reported as suspicious for two reasons: (a) the quality and standardization of data collection vary, and (b) even if useful data are gathered for in-house use, it may be modified for various reasons for presentation to those outside the company.

Developers and experts interviewed for this research often kept some kind of records of the development times for previous projects. However, many of the developers did not believe the historical data could be used because the project(s) had unique factors which other projects would not have. This same caution was voiced so frequently that it may be necessary always to expect unique factors. Other developers reported .at data kept was incomplete or difficult to access.

Although most developers agree that historical data are useful for predicting CBT development costs, there is no standardized systematic procedure for gathering and applying data when making estimates. In addition, some developers seem reluctant to use the data available because "that project was unique." In addition to these methods, a few organizations are beginning to develop CBT development cost models. These are discussed in Part Three.

CBT Development Factors

Many factors must be considered in estimating CBT development costs. Experts have a wide variety of opinions about which factors have the most significant impact on CBT development times, but there is little supporting evidence for these beliefs. This section of the report discusses factors commonly believed to significantly affect development times.

Gery (1986) identified 37 factors which impact development time and grouped them into four major categories:

- Courseware variables
- Technical variables
- Human variables

- Other variables

Under the category of courseware variables she listed, for example, the nature and complexity of the learning material, the level of the learning objectives, the instructional design strategy, and the nature and frequency of the interactivity. In the category of technical variables are the authoring tools capabilities and limitations, and the productivity tools available. In the category of human variables she listed the number of people in the development team, the experience of the team members, and the number of projects the team has worked on together. Finally, in the category of other variables are such factors as the availability, nature, and quality of existing training materials, and the availability of a graphics library.

Gery suggests the user assign weights to each factor and plot relevant factors on a scattergram to determine approximate development hours. For example, if most of the variables plotted are in the "low" end of the scattergram, Gery estimates that development may take between 85 to 150 hours to produce one hour of instruction. Gery defines an hour of instruction as the amount of time "at the computer taking a course that is essentially linear in nature, including conditional feedback, and restricts the use of conditional branching to review segments" (p. 37).

Most experts agree that identifying and assigning weights to variables are essential prerequisites for deriving accurate courseware development estimates. However, Gery's method has yet to be validated. Translating ratios of variables into accurate cost estimates may prove to be a challenge for courseware developers but still will not yield common results across developers unless weights and costs are standardized for types of courseware. The factors most commonly cited by CBT experts and developers as affecting development time are discussed below.

Instructional Strategies. Instructional strategy may be defined as the design for implementing the elements of instruction (e.g. motivation, presentation, feedback) for the lesson content. Examples of general types of instructional strategies include simulation, drill and practice, and inquiry.

The type of instructional strategy used in a CBT lesson is the most frequently mentioned major factor in determining the amount of development time required. For example, an instructional strategy that merely presents text on a screen and allows only forward and backward movement would require substantially less development time

than a full simulation of a complex mechanical device.

A study conducted by Gailey (1973) compared development times required to produce lessons using two different instructional strategies. Ratios for development time to student on-line time were identified. The ratio for writing simulations was 175:1, and the ratio for writing tutorials was 114:1.

Kearsley, Wilson, Gailey and others believe that instructional strategies significantly impact CBT development times. Other experts argue that there is currently insufficient evidence to validate this claim. Although the reported experience of many CBT developers indicates that instructional strategy is a significant development factor, there is little evidence on the quantifiable impact of this factor on development times.

Instructional Features. Many experts believe that instructional features are a major factor determining the amount of time required to develop CBT. Examples of instructional features include text, color graphics, animation, glossaries, video, audio, speech recognition, input devices, etc. If the features are embedded in the authoring system, they generally do not substantially increase development time. However, features which must be programmed do increase development time.

Kearsley (1983) states that different instructional features cost different amounts--and that the number and complexity of the features, increases the development time and costs. For example, a screen with video, color graphics, text with audio and touchscreen input generally requires more development time than a screen with text and keyboard input.

Content Complexity. Content complexity may be defined as the technical difficulty and sophistication of the instructional content. For example, explaining relationships between components of an electronic circuit is more complex than identify the names of the components. Hence, the technical complexity of instructional content is frequently mentioned as having a significant impact on the amount of time required to develop CBT. There are two reasons for this:

1. Unless subject matter experts (SMEs) are also experts in CBT design and development, either courseware designers must gain proficiency in the technical content or the SMEs must learn instructional design skills.

2. In technical training situations, there is often such a broad range of student familiarity with the content

that courseware must be designed to accommodate both students who have little or no previous content knowledge and those who have some previous knowledge.

Purchaser/Developer Politics. "Political" factors were mentioned by many developers and experts as affecting CBT development costs (Kearsley 1985; Gery 1987; Mikos, et. al. 1987,). Examples of political factors mentioned are: several managers involved in decision-making and review; a client inexperienced in CBT who demands high involvement in development; wavering client commitment to the project; unavailability of assigned subject matter experts; and unstable client interface. Such factors affect any training development project with delays and changes. The cost impact of these factors is greater in the computer based medium because revisions are more expensive, especially if tasks such as video production or coding have been completed when major changes are requested.

Authoring Tools. All courseware is translated from a writer's ideas to computer code through some type of authoring tool. The authoring tool used for courseware development is also believed to be a major contributing factor to CBT development time. Fairweather and O'Neal (1984) divided authoring tools into four categories:

1. programming languages
2. authoring languages
3. authoring systems
4. hybrid authoring systems

Each of these tools has different capacities and ease of use. The more powerful tools such as programming languages and authoring languages use computer code and require programming expertise, making them more difficult to use. Authoring systems generally use menus rather than code, and although they are less powerful, they are easier to use, because they allow simplified data input. Hybrid authoring systems, are designed with the intention of incorporating the best features of programming languages, authoring languages, and authoring systems. There does not appear to be any evidence corroborating claims that hybrid authoring systems provide the best features of other systems without their disadvantages.

A systematic study on the effects of authoring tools on development times was conducted by Hillelsohn (1984). His study was an empirical comparison of the time it took to develop a "benchmark lesson" on five commercially available authoring systems. Although Hillelsohn found it difficult to compare the quality of the lessons produced, he did find differences in development times required by developers using different authoring tools to produce the same predesigned, storyboarded lesson. Of the two developers (out of five) who completed the lesson within one day, the ratio of development time to instructional hours was 60:1 for one system and 96:1 for the other, not including instructional design, review, or revisions.

Buck (personal communication, January, 1987) said production times decrease in relationship to the extent an authoring tool supports both the design and development functions, because such tools facilitate rapid communication between subject matter experts, instructional designers, and programmers. New generations of authoring tools are incorporating design functions, such as storyboarding, that used to be performed off-line.

The impact of authoring tools on development time must be systematically considered by identifying, matching, and rating available tools for particular applications. Although there is insufficient data to accurately predict the impact of design tools on production, clearly they are a relevant factor that must be considered.

<u>Author and Programmer Experience</u>. CBT staff experience is another factor which is widely believed to have a major impact on development time. Generally, more experienced CBT authors and programmers need fewer hours for development.

Avner (1979) studied the effects of author experience and type of pedagogy on production rates. He compared authors who had more than two years of experience with the PLATO authoring language with authors who had less than one year of experience. The range of development times required by experienced authors was markedly lower than that required by inexperienced authors.

Grimes (1975) studied the productivity of student and staff programmers at the University of Illinois. Although there were no appreciable differences in student and staff performance, programmers were significantly more productive in the second year than they were in the first year of the study. Individual experience probably accounted for some portion of the productivity increases, and the development of code template and programming subroutines probably accounted for another portion.

Avner, Smith and Tenczar (1984) reported results obtained from the longitudinal observation of 143 independent CBT production groups. They reported that production teams which have worked together on past projects had better internal communications than newly organized teams. Improved communications could be a factor in increased production. Buck and Gillespie (1985) reported that experienced authors required 100 hours to develop one hour of tutorial instruction, whereas inexperienced authors required 200 hours.

Although the relationship of author and programmer experience on CBT development times has not been quantified, the above studies suggest that experience does affect courseware development times and costs.

Production Schedule. The amount of time allotted for the production of the courseware has also been cited as a factor in CBT development times. Avner, et al. (1984) used data from 143 independent CBT production projects to identify qualitatively which factors influenced production efficiency and quality. Four factors were identified as the best predictors of development time. These included: (a) production deadline, (b) software authoring tool, (c) media experience, and (d) instructional methods experience. The authors reported that production will require all the time allocated if the production team knows the deadline. They cited a situation where the use of an authoring tool allowed courseware to be produced more quickly but increased efficiency was not seen. The reason suggested for this was that the work "artificially stretched out to meet the overly generous ... deadline and that the 'free' time was being used on other materials that took longer than the 'predicted' time" (p. 86). However, a Avner noted (personal contact, June, 1987) that there is a minimum amount of time required to complete a lesson, and if that time is reduced, the quality of the courseware suffers significantly.

Brooks (1982) expressed another view on factors influencing software development that is relevant because of the similarities between software and courseware development processes. He claims that more software projects have gone awry for the simple lack of calendar time than for all other causes combined and that adding personnel to a project behind schedule is seldom effective. He said there are five reasons why this problem is so common. First, current techniques of estimating are poorly developed. More seriously, they reflect the unvoiced assumption which is quite untrue: that all will go well. Second, estimating techniques falsely confuse effort with progress, hiding the false assumption that people and months are interchangeable. Third, software project managers, in their rush to meet a deadline, are likely to compromise the quality of the software. Fourth, schedule progress is poorly monitored. Techniques proven and

routine in other engineering disciplines are considered radical innovations in software engineering.

The fifth reason Brooks cited is that when schedule slippage is recognized, the natural and traditional response is to add manpower without allowing for increased training and intercommunication. Adding manpower can only speed up production if the tasks can be partitioned in such a way that no additional communication is required among workers. But, if communication is needed, as is almost always the case in courseware development process, "Adding more men then lengthens, not shortens, the schedule" (p. 19).

Government Requirements. Other factors which impact CBT development times relate to government requirements for CBT development and contract reporting. Government contracts require that CBT developers use the Instructional Systems Design (ISD) process for developing courseware. They may also require that developers extensively document their adherence to the ISD process. Using and documenting ISD can increase the time and costs associated with CBT development. According to R. Foshay (personal communication, January, 1987), this ISD emphasis requires that a sharp distinction be made between estimates made for government and commercial clients. Government CBT contracts typically require several types of reports, such as analysis reports, design reports, and final reports. On the other hand, commercial CBT contracts may not require strict adherence to an ISD process and typically have formal delivery requirements for only the finished courseware. Reports and communication are typically more informal, structured only as needed for effective communication. The reduced adherence and reporting requirements of commercial CBT contracts can significantly reduce development time and costs.

Although there have been few studies to quantify the factors affecting courseware development time, eight factors were most often cited by experienced developers. They are, complexity of content, instructional strategies, instructional features, purchaser/developer politics, authoring tools, developers' experience, production deadlines, and government reporting requirements. In the next section seven completed CBT contracts are examined in light of the cost factors and issues raised above.

Contract Examples

In interviews with CBT development project managers conducted for this report, anecdotal information was gathered on completed contracts. Information was requested on the cost of the project, the hours required to produce each unit of courseware, and the factors which were judged by the project manager to be responsible for

discrepancies between projected and actual costs. These factors may not have contributed most to the cost, but were major in the cost budget overrun. (This explains why these factors are not the same as the eight from the literature search.) A summary of these findings is presented in Table 1.

Of the seven projects, five were over budget, one was within budget, and one was not tracked closely enough to determine the accuracy of the original estimate. Of the five over budget, two reported that they were near the budgeted amount, while the remaining three were substantially over budget. In one case, the project manager reported that the project was completed within budget, however a change comment from the client revealed that there was substantial evidence that the project cost the company more than budgeted. (The calendar time required to complete the project was substantially longer than planned, and a company representative indicated to the client that the project should have been bid much higher.)

Estimated ratios of development time to instructional time for these contracts reportedly ranged from 55:1 to 351:1. Actual development time figures were not available for five of the contracts, either because the information was considered proprietary (contract A), or accurate information was not recorded (contracts B-E). In some cases, development time was not tracked, or it was only tracked for one member of the team.

Of the four companies interviewed, only two have developed systematic methods, incorporating data bases from past development efforts, for estimating development costs. One company uses a proprietary in-house model which incorporates three categories of information: ISD parameters, historical data, and development variables. The other company reported that the estimate for the second project was much closer to the actual amount when they used historical data and a Work breakdown Structure. Project managers from both companies indicated that they now prefer to bid the analysis phase separately from the rest of the contract.

The remaining two companies have not developed systematic methods for making CBT cost estimates. The company which developed contracts B and C above, estimates project costs by determining labor costs for the project manager. Other labor and production costs are absorbed by various departments, and there is no comprehensive development budget or tracking of the projects. This group has no specific program plans to improve estimating procedures, but did state they see a need for more operations research. The remaining company uses a development ratio of 200:1 to make estimates and has no plans to institute data collection. They do plan to include a greater margin for error in future contracts.

Table 1

Anecdotal Information From CBT Contracts

Contract	Cost	Developer's CBT Experience	Development Hours Per Hour of Instruction	Final Cost	Reported Cause of Overrun
A	$150,000	5 years	145:1[a]	Over budget	High client learning curve Weak RFP Client staff turnover Content changes
B	$ 36,000	4 years	100:1[a]	Over budget	None reported
C	$ 24,000[b]	4 years	100:1[a]	Unknown	None reported
D	Proprietary	20 years	55:1[b]	Over budget by 100%	Change in project scope Development compexity Staff inexperience
E	Proprietary	20 years	147:1[b]	Within budget	Content changes Reviewer delays
F	$2,500,000	6 years	170:1	Over budget	Staff inexperience Poor management Poor communications Poor reviewer interface
G	$722,000	16 years	351:1[a]	Over budget	Increase in project scope Content complexity Reviewer delays Technical complexity

[a] Amount billed to client. Actual figures may be substantially higher, but detailed information is either proprietary or unknown.

[b] Project manager salary only. Other personnel were in different cost centers; their labor was not tracked.

The most common factors cited as responsible for discrepancies between estimated and actual costs were reviewer changes and delays, staff turnover, content complexity, and development complexity. Several developers noted that vague RFPs resulted in changes in project scope.

The more experienced developers in this organization indicated that the amount of development experience of the CBT team is a key factor determining production efficiency. This is particularly true if past experience is systematically incorporated into improved modes of functioning in future contracts.

Of this sampling of four CBT developers, only half are beginning to develop systematic methods for estimating development costs. The range of development hours are suspicious because developers did not track development hours completely or reported the ratio for the least expensive hour produced. From these examples, it is evident that companies deal with the difficult problem of estimating costs in a variety of ways, from ignoring the problem to trying every possible method to improve practices. Even for the organizations willing to invest in developing systematic project tracking and estimation methods, each company must start from scratch, analyzing its own historical data and determining cost factors. These companies, and others like them, would greatly benefit from the development of a universal CBT costing model to assist in this difficult process.

Summary

In this part of the report, current practices and problems associated with estimating CBT development costs were examined by searching available literature, contacting CBT professionals, and gathering anecdotal information on completed CBT projects. Most experts agree that it is difficult to estimate accurately CBT development costs because:

- The medium is inherently complex.
- There are currently no standardized methods for collecting and reporting development data.
- There is currently no effective or standard method of measuring courseware.
- There are many factors impacting development which have yet to be assigned standard definitions and weights.

- There are no standard measures of courseware quality.

- Contracts are typically bid prior to a complete front-end analysis, so estimates must be made without full knowledge of project scope.

There is a critical need for accurate, standardized CBT cost estimating tools so that: 1) purchasers of CBT have a basis for discriminating and justifying between proposals, and 2) developers have a basis for projecting realistic development times and costs so they can manage projects to stay within pre-established limits. Some developers feel it is premature or inadvisable to try to create a standardized tool for two reasons: 1) the field of CBT is too young and still undergoing rapid change and evolution, making it difficult to create a comprehensive tool, and 2) such a tool might introduce a constrained model of the courseware production process which might stifle innovative development approaches.

Current estimating practices are so ill-defined and haphazard that many mistakes are being made, and purchasers and developers of CBT are struggling to find better estimation methods. Most experts agree that there is a need to define and remove the obstacles to accurate cost estimating so the contract bidding and procurement process is based on a realistic foundation.

A primary difficulty in estimating costs is that developers often do not collect data on projects and, therefore, have no database to use for future cost estimates. Or, if they do collect data, it is not collected or reported in a standardized manner from one developer to another. Also, because the government does not require developers to report CBT development data, there is no external incentive to do so.

There is some indication that some developers either deliberately underbid to get into the business or apply minimum development times assuming the everything will go perfectly.

In addition, there is no universal method for measuring CBT. The instructional hour is most frequently used, but there is no agreement on a standard definition of the hour as a measurement unit. There is evidence that the hour, as well as other units such as screens, interactions, objectives, and lessons, is insufficient by itself as a method of measuring CBT. Each of these methods addresses only the quantity of courseware being measured, and experts suggest that the quality of courseware must also be measured.

If development time ratios are to be of any use, those using them must know whether the courseware they are developing is similar in features and quality to the courseware the averages were based on. Development time averages typically range from 100:1 to 400:1. Gery (1986) states,

> Such imprecise estimates are nearly useless. Even if these broad ratios are "accurate" or generally applicable, their use produces essentially indefensible figures. Try justifying a budget based on something as vague as an "industry average" when demands for detailed explanations or pressures to reduce an "unacceptably high" estimate are being put to you (p. 31).

Because the type and complexity of courseware varies widely, it is essential to have a clear definition of the critical factors which impact CBT development time to make accurate cost estimates and have a basis for comparing courseware. A number of factors were discussed, including instructional strategy, content complexity, instructional features, politics, authoring tools, author and programmer experience, and others. There are many beliefs about which factors are critical in making time and cost estimates, but there is little research to support the beliefs of about the relative importance of each factor. There is indication that consideration of project complexity factors can improve development estimates.

In Part Two of this report, information was systematically collected from almost 200 CBT developers concerning CBT development estimation methods, CBT measurement, and development cost factors.

PART TWO: A SURVEY OF DEVELOPERS

In Part One of the study, we attempted to identify the issues which affect CBT development cost estimates. Part Two of the report contains a description of a survey conducted to gather information from almost 200 CBT developers. The main purposes of this survey were to: (1) identify the ways in which CBT developers currently estimate costs, (2) collect data on average CBT development times, and (3) rank order the factors CBT developers report as having the most influence on CBT development costs.

Based on the information obtained in Part One, three questions were asked about the current methods CBT developers use to estimate development costs. The three questions were:

- What method do developers use for estimating CBT development costs?
- What are the actual development times required to produce courseware?
- Which factors have the greatest influence on CBT development times?

Method

Subjects

The participants in the survey work for organizations which produce computer-based training. Three lists were used to identify potential subjects: (a) the membership list of the Association for the Development of Computer-based Instructional Systems (ADCIS), (b) "Directory of Courseware Vendors" published in Data Training (1986, March), and (c) a list of CBT developers who were known to the researchers and who had volunteered to participate in the survey. Subjects on more than one list received only one survey. The final mailing list consisted of 87% ADCIS, 10% Data Training, 3% volunteers. From these lists the following criteria were used to select 1,100 individuals to receive the survey.

1. Only organizations on the North American continent were included.
2. Nondevelopers, such as libraries and professional organizations were excluded.

3. Generally, only one subject per organization was included.

4. Only major universities were included; small universities and colleges were excluded.

Procedures

A questionnaire was designed to collect information about the characteristics of CBT developers, how they cost CBT, average development times, and factors they believe affect development costs. Appendix B contains a copy of the questionnaire.

The questionnaire, along with a cover letter explaining the study was sent to 1,100 potential subjects. As an incentive to participate, subjects were offered a summary of the results of the study. Questionnaires were anonymous, and subjects desiring the results sent in a separate postcard. A second mailing was made to 300 private developers who had not returned postcards four weeks after the initial mailing.

To assure responses and facilitate the ease of gathering information, the questions posed did not require subjects to refer to actual records. The questionnaire consisted of 38 multiple choice and 15 open-ended questions. This questionnaire was designed for quick and easy responses while still allowing open-ended responses when answers could not clearly be anticipated.

Data Analysis

Researchers selected 20 questionnaires and analyzed the open-ended responses to establish categories for coding the responses. By informal agreement, the researchers established consistency in coding the responses. The emphasis of the data analysis was mainly on descriptive statistics for the population of interest, not on the strength of the relationships among variables.

Results and Discussion

Population Characteristics

The results of the survey are presented in the following order. First, data are presented that describe the general characteristics of the population that responded to the survey. Second, data are presented related to each of the three general questions. Data are generally reported here as percentages of respondents answering affirmatively to the questions. For many questions, the percentage sums are more than 100% because the categories are not mutually exclusive.

Of the 1,100 questionnaires mailed, 211 were returned. Forty of the returned questionnaires were so incompletely answered that they were discarded. The total usable sample was 179. This represents a return rate of 16%.

The respondents categorized their organizations in the following ways:

- private organizations 67%
- academic institutions 22%
- government agencies 7%
- other categories 4%

The mean number of employees involved in CBT development was 21; the median was eight. Private organizations had a higher mean number of employees (23.7) than academic institutions (14.7).

The mean years of experience was six years. Academic institutions had more CBT experience (8.0 years) than either private organizations (5.4), or government agencies (4.6).

CBT organizations reported that they develop courseware for the following markets:

- in-house use 44%
- custom 42%
- off-the-shelf 37%

Sixty-seven percent of the respondents indicated producing technical courseware and 41% indicated producing academic courseware. Private organizations reported their courseware contained technical content more than twice as frequently as academic organizations.

Courseware can be developed with either a computer programming language such as Pascal, or with a language or system created for authoring courseware. Subjects reported using the following tools:

- authoring language 61%
- authoring system 54%
- programming language 46%

Methods of Estimating Development Costs

Information was sought about CBT organizations' methods and accuracy in estimating CBT costs. The researchers also compiled the participants' recommendations for improved bidding methods. The data relating to cost prediction accuracy are presented first.

Cost prediction accuracy. Respondents described their ability to predict costs by marking one of four accuracy categories. Table 2 lists the categories and the responses given by all organizations. Only 11% of all organizations claimed they are able to estimate the cost of CBT within 5% of cost. More than a third (38%) claimed their estimates are accurate to within 10% of actual cost. Twenty-two percent of organizations claimed their estimates are accurate to within 20% of cost, while 29% of the organizations reported actual costs exceeded their estimates by 20% or more.

Table 2
Cost Prediction Accuracy and Group Responses.

	n	Accuracy of estimates			
		Within 5%	Within 10%	Within 20%	Exceeds 20%
Percent of organizations	157	11%	38%	22%	29%

Respondents identified the reasons for the divergence of their estimates from actual project costs. The question was open-ended, and the respondents' answers were categorized. The most frequently mentioned reasons were:

- weak RFPs/changes in scope & unexpected revisions from the client 36%
- failure to estimate the number of man hours required 12%
- complexity difficulty to gauge 10%
- maintaining quality control 5%
- client, subject matter expert or needed materials unavailable 4%
- staff turnover 3%

Many respondents (35%) gave unique reasons. Some examples of these are: "substandard work requiring several revision cycles," "complexity of the design increased during development," "interactive video graphics," "changing criteria used by in-house reviewers," "problems with hardware or software," and "unpredictability of productivity of staff with no previous CBT experience."

The most frequently cited reason CBT developers gave for inaccurate estimates was the tendency for the client to change the scope of the project or to request unanticipated revisions.

Unit of Measure. Respondents identified the unit of measure used to estimate development costs.

- 27% use the instructional hour.
- 15% named other single measures, such as the complexity of the content or the number of interactions.
- 9% use the number of lessons as the unit of measurement.
- 7% use the number of screens required.
- 42% reported that they use a combination of units in a mathematical formula. For example, some use the number of screens multiplied by the number of interactions, others use the number of instructional hours multiplied by the degree of complexity.

Respondents named the advantages and disadvantages of the unit of measure they use when estimating costs. The responses of those using the hour, lesson, and screen as units of measure were analyzed. Forty-two percent of the respondents reported they used a combination of ways to measure courseware. Their comments on advantages apply only to their unique combination and therefore were not tabulated. Table 3 lists the advantages and disadvantages named by each group for three single types of measurement.

Tasks Included in Estimates. When estimating the cost of the development process, developers do not all include the same tasks. Table 4 shows the percentages of respondents who include possible tasks in their estimates. For example, although 86% included writing lessons in their estimates, only 60% included management time. The implication is that cost estimates made by different organizations bidding on the same development project must be compared carefully because they may not include the same tasks and therefore may not result in the same product.

Distinguishing between Types of Courseware. Sixty-one percent of the organizations replied that they distinguish among different types of courseware when they estimate the the cost of CBT projects. Of these organizations that do distinguish among the types of courseware to be produced:

- 26% use the type of instructional strategy (e.g., drill and practice, tutorial, simulation) required.
- 25% use the other single dimensions of discrimination such as the complexity of the content, or the total number of graphics needed.
- 11% use the types of courseware based on the type of instruction required. For example, they estimate cost differently for computer-managed instruction, computer-assisted instruction, and interactive videodisc.
- 38% use a combination of factors when costing CBT projects. For example, the combination of instructional strategies times the number of screens plus the number of graphics.

Respondent Recommendations. Currently, there is no industry-wide standard for measuring courseware. Respondents were asked if they would like to see an industry standard for measuring courseware and the reasons for their position. The responses are listed below in Table 5.

Table 3
Advantages and Disadvantages of Units of Measure

Unit of measure	n	Advantages	Disadvantages
Instructional hour	44	26% simplicity 9% accuracy 13% client preferred 13% uses historical data 9% concrete	12% too simplistic 32% inaccuracy 26% does not reflect complexity
Lesson	14	40% uses historical data	20% inaccuracy 20% does not reflect lesson size
Number of screens	12	75% accuracy	63% does not reflect complexity

Table 4
Frequency of Tasks Included in Cost Estimates

Task	Frequency included
Writing lessons	86%
Front end analysis	82%
Revisions	79%
Developing graphics	78%
Programming lessons	77%
Learning content	68%
Programming routines	62%
Management time	60%
Formative evaluation	54%
Meetings	52%
Secretarial support	45%
Summative evaluation	42%
Video production	35%
Computer operations	34%
Technical reports	23%
Computer down-time	11%

Note. n = 166.

Table 5
Percent Favoring an Industry Standard for Measuring Courseware

Responses	Percent
Yes, would provide a basis for comparisons	17
Yes, would provide fairness	5
Yes, would be effective	5
Yes, other reasons	32
No, not possible	22
No, would not be useful	8
No, other reasons	11

Note. n = 146.

Fifty-nine percent of the respondents favored a standard while 41% did not. Many of the respondents gave reasons for their position which were difficult to categorize. Some examples are: "Yes, it would help when going to different courseware houses to determine what you are really getting," "Yes, but I doubt it will work," "No, there is too much variance in CBT style." "No, It would enforce a too constraining model of the courseware production process, stifling innovative approaches that might make conventional cost-estimating measures irrelevant."

Respondents who favored an industry-wide standard for measuring courseware were asked to suggest methods of measurement that would most accurately reflect true development times. Only one percent recommended the instructional hour as an accurate unit of measure. Of those who replied (n = 78):

- 36% recommended the use of a mathematical formula.
- 10% the number of development hours.
- 6% the number of interactions.
- 5% the number of screens.
- 3% the number of lesson.
- 1% the number of instructional hours.
- 21% unique units.

Eighteen percent reported that they did not know a better method of measuring courseware.

Some examples of suggested mathematical formulas are: "level and number of interactions," "number of screens and graphics per lesson plus a scaling factor for animations, video, and audio," "number of interactions plus the time for average person to go through the lesson," and a formula that would include "instructional hours, complexity of the lesson, and the number of graphics."

Forty-two percent of the respondents already use a combination of units to measure courseware, so it is not surprising that 36% recommended that some kind of formula be used to measure courseware.

Respondents suggested their ideal methods of bidding CBT contracts.

- 16% favored a cost plus method.
- 14% favored bidding by phases of the project.
- 8% recommended bids based on time and materials.
- 6% suggested using a fixed price.

Nine percent replied they did not know of a better method of bidding CBT contracts. Nearly half (47%) of the respondents gave other suggestions, none of which exceeded 5% of the replies. Some examples of these are: "Cost comparisons to other media;" "Ideally, if we could quantify interaction level, we could bid on the whole project, rather than re-budget after development;" "(1) Set milestones with signoffs, (2) penalize either party for flagrant violation of deadlines -- must be stipulated up front. (3) both parties have project managers who get bonuses for timely completion without incurring additional expenses."

Actual Development Times.

Respondents answered questions about the actual number of hours required to develop a unit of CBT. The first part of the question asked "What is the range of hours that your organization actually requires to produce a unit of courseware?" Respondents were not required to use actual records. The results are separated below by the units of measurement used by the organizations.

Respondents reported the range of hours required to develop a unit of courseware. The questionnaire did not ask for hours for different types of courseware, so all types are included together in the figures. For example, when reporting the minimum time, the respondent may have been reporting the development time for drill and practice courseware while the development time reported for the maximum time may have been for simulation courseware.

The times reported varied widely for both the minimum and maximum times. For example, the lowest time reported was one hour, the highest time was 4,000 hours. The results are reported in Table 6. The analysis of the data was restricted to looking at two units of measurement: the instructional hour and the lesson. Although respondents reported in other units, these units were not used often enough to allow for statistical comparisons. Because the respondents reported a range, the mean of all minimums and the mean of all maximums were computed. The table shows that these reported means are in line with industry averages commonly cited.

Table 6
Means of Minimum and Maximum Development Times

Unit of measurement	n	Mean minimum	Mean maximum
Instructional hour	56	140 hours	316 hours
Lesson	13	68 hours	351 hours

Thirty-four percent of the respondents reported development times for some of their courseware in the range of 400-999 development hours per hour. Twenty-seven percent reported development times for some of their courseware in the range of 1000-4000 hours per hour. Private, academic and government organizations were all represented in these groups. Some developers gave reasons for the high development time. The most commonly cited reasons given include:

- "Highly simulated course."
- "Client requested changes too late in the game."
- "[Academic developers] have little need for cost control."
- "Initial lessons take longer, subsequent lessons take less."

- "No good development model."
- "Changes in product that course describes."
- "Interactive videodisc courseware with novice developers."
- "Vague specifications."

Information was collected on the number of hours required to perform a series of development activities for four different types of courseware. The results of those who reported development times in instructional hours are presented in Table 7.

Development times increased as the type of instructional strategy used in the courseware increased in complexity. Although the number of hours increased with the instructional strategy, the reported times of the activity remained a fairly constant percentage of the total development time. Instructional design, for example, accounted for an average of 14% of the total hours for developing an instructional hour of drill and practice courseware, 12% for tutorials, 13% for simple simulations, and 12% for complex simulations.

The average reported times for each development activity were about the same across all four types of courseware. A noticeable exception to this was the percentage of time required for writing the programming code for complex simulations. It accounted for 30% of all the time required to develop a complex simulation compared to approximately 23% for the other types of courseware.

The reported development hours required for the four types of courseware are listed in Table 8. Totals are given only for hours and lessons of courseware because other units were not reported often enough by the respondents to allow comparisons. In general, development times tend to increase as the instructional strategy becomes more complex. For example, when courseware is measured in hours of instruction, drill and practice type instruction averaged 164 development hours, and complex simulations averaged 343 development hours per instructional hour.

Table 7
Mean Hours and Percent of Development Time per Task When Unit of Measure is the Instructional Hour

Development activity	Type of courseware: Drill & practice	Tutorial	Simple simulation	Complex simulation
Instructional design	22.5(14%)	28.4(12%)	30.3(13%)	39.7(12%)
Writing content	33.9(21%)	36.8(16%)	40.4(17%)	60.3(18%)
Writing programming code	37.9(23%)	50.8(22%)	56.3(23%)	101.5(30%)
Creating graphics	14.0(9%)	18.0(8%)	30.5(13%)	42.1(12%)
Storyboarding and producing video	15.0(9%)	39.4(17%)	22.1(9%)	19.3(6%)
Reviewing and implementing in-house revisions	17.7(10%)	27.3(12%)	31.8(13%)	44.0(13%)
Reviewing and implementing client requested revisions	11.7(7%)	19.0(8%)	18.5(8%)	24.8(8%)
Other	11.3(7%)	9.3(4%)	12.1(5%)	11.3(3%)
Totals	164.0 (100%)	229.0 (99%)	242.0 (101%)	343.0 (102%)

Note. n = 52.

Table 8
Mean Total Development Times of Four Types of Courseware

		Type of Courseware			
Unit of measure	n	Drill & practice	Tutorial	Simple simulation	Complex simulation
Hour of instruction	52	164	229	242	343
Lesson	11	132	198	167	299

Cost Factors

In an effort to validate factors affecting development costs, 26 factors were listed, including many of Gery's, and CBT developers were asked to rate the effect of these factors on the cost of developing CBT. Table 9 lists the means of responses for each of the factors. The factors are listed in order of effect on cost. The highest possible rating was five.

The complexity of the instructional design strategy received the highest mean rating. This is consistent with the data reported earlier that development time for a complex simulation is greater than for a drill and practice, especially for the tasks of writing and programming. In addition, at least 15% of the respondents use type of instructional strategy as a basis for estimating costs.

The nature and complexity of the content was the second highest rated cost factor. One interpretation of this result is that designing and developing a complex technical lesson for a student who lacks basic knowledge of the technical field in question is more time consuming than developing courseware which treats less complex content. Technical complexity may also affect development time because designers and developers must spend extra time to become familiar with content.

Three factors associated with instructional features of the courseware were ranked fourth, fifth, and sixth. They were, respectively, complexity of screen interactions, complexity and volume of graphics, and conditional branching complexity. Three others (complexity and volume of video, response analysis complexity, and extent and complexity of the feedback) appeared more towards the middle of the list.

Table 9
Ratings of Factors' Effects on Cost

Factor	Mean
1 Complexity of instructional design strategy	4.54
2 Nature and complexity of content	4.39
3 Client's demand for revisions	4.08
4 Complexity of screen interactions	4.06
5 Complexity and volume of graphics	4.04
6 Conditional branching complexity	4.02
7 Development team's experience producing CBT	4.01
8 Author's experience with instructional design	3.94
9 Complexity and volume of video	3.94
10 Programmer's experience with programming language	3.93
11 Author's experience with the content	3.87
12 Interfacing CBT with another system	3.87
13 Capabilities of authoring language or system	3.85
14 Turnover of CBT team and client	3.78
15 Response analysis complexity	3.72
16 Extent and complexity of the feedback	3.71
17 Type of behavior to be learned	3.70
18 Amount of client control over the project	3.68
19 Ease of use of authoring system or language	3.66
20 Availability of reusable programming routines	3.66
21 Project manager's experience	3.55
22 Client's available time & commitment to project	3.54
23 Number of people involved in the process	3.52
24 Availability of text processor in authoring system	3.29
25 Repetition of development tasks	3.25
26 Availability of in-house subject matter expert	3.00

All of the factors were rated as having a moderate (3.00) to major (5.00) effect on development cost, probably because the list was limited to factors already believed to affect cost.

Of all factors relating to developer experience, "development team's experience producing CBT" was rated highest (seventh). If one interprets this as meaning the team's experience working together as a team on previous CBT projects, then team experience is ranked higher than individual experience. This might be because of the collaborative nature of CBT, where a premium is placed on establishing procedures and effective communication among team members.

Respondents did not rate the cost effect of the authoring system highly. This unexpected result could be influenced by several considerations. First, developers may be able choose authoring tools best suited to particular applications, avoiding the inefficiencies of ill-suited tools. Second, programmer's experience with the language was higher (tenth) than the ease of using the system. This may mean that an experienced programmer's ability to exploit the strengths of a system is sometimes more important than the system itself. Finally, the effects of the system may be confounded with other factors on the questionnaire. For example, the cost effect of complexity of screen interactions (fourth) may be a function of the ease of coding interactions with the system being used.

Some respondents named cost factors that were not on the list provided. The most commonly added factor was whether a clear and complete specification was available at the beginning of the contract (for example, "client knows what really wants at beginning" and "completeness of initial specifications").

Other factors mentioned include:

- "urgency"
- "amount of customer review"
- "size of team"
- "staff turnover, training costs"
- "stability of authoring system"
- "use of outside programmers"
- "setting standards... and adhering to them"
- "availability of appropriate tools"
- "capabilities of local IVD firms"

Results and Discussion by Groups

The results were subjected to statistical analysis to determine if there were any significant differences between groups distinguished by criteria of special interest. The groups examined included:

- developers with 0-2 years experience vs. those with 3 or more years experience,
- academic vs. private organizations, and
- accurate (within 10%) vs. inaccurate (over 10%) estimators.

Significant differences are reported by the criteria defining the groups; non-significant differences are not reported unless they were counter to expectations. Possible reasons are given for the differences found, based on the literature search and interviews with developers.

Years of Experience.

Development Times. Several sources cited in the resource review said that development time decreases with experience. However, the data do not show any significant difference between the times it takes each group to develop a unit of CBT. Additional investigation into this surprising finding is recommended, but a possible cause can be suggested. Although more experienced developers may have developed time-saving tools and techniques, the inclusion of additional courseware features may negate the savings resulting from the use of those tools. For example, more experienced developers may be adding features such as glossaries and help screens, or developing more complex lessons than less experienced developers.

Cost Prediction Accuracy. A chi square test revealed significant differences in cost estimation accuracy based on the organizations' years of CBT experience (Chi square (3) = 9.4, $p < .05$). Table 10 lists the results.

Table 10
Years of Experience and Cost Prediction Accuracy

		Estimates			
Experience	n	Within 5%	Within 10%	Within 20%	Exceed 20%
0-2 years	34	15%	32%	9%	44%
3 or more years	115	10%	42%	26%	22%

Half of all developers reported that they were able to predict costs within 10% (47% of inexperienced and 52% of experienced). However, almost twice as many inexperienced developers were likely to make estimates that are off by 20% or more.

Measurement Methods. Of the developers who use composite means to measure courseware (n=110), significantly more had three or more years experience (82%) than had up to two years experience (18%) (chi square (1) = 3.69, $p < .05$).

Tasks Included. Table 11 shows that more experienced organizations more frequently include "management time" as part of their cost estimates than do less experienced organizations (chi square (1) = 5.0, $p < .05$). Table 11 also shows that more experienced organizations more frequently include "developing graphics" as part of their cost estimates than do less experienced organizations (chi square (1) = 4.4, $p < .05$).

Table 11
Years of Experience and Differences in Tasks Included in Estimates

		Task	
		Project Management	
Experience	n	Included	Not included
0 - 2 years	37	43%	57%
3 or more years	122	66%	34%
		Developing graphics	
		Included	Not included
0 - 2 years	37	65%	35%
3 or more years	122	83%	17%

Academic vs. Private

Accuracy. Table 12 shows that more private organizations (44%) reported their cost estimates are within 10% of actual costs than academic institutions (20%). More academic institutions (46%) reported that their costs exceeded estimates by more than 20% compared to private organizations (23%) (Chi square (3) = 10.5, $p < .05$).

Table 12
Private and Academic Institutions and Cost Prediction Accuracy

		Estimates			
Institution	n	Within 5%	Within 10%	Within 20%	Exceeds 20%
Private	107	9%	44%	22%	23%
Academic	35	17%	20%	17%	46%

Academic organizations have higher percentages of respondents in both the most and least accurate ranges. The ability of more academic than private organizations to predict costs within 5% may be explained

by the fact that academic institutions have had more years of experience producing CBT and that the CBT they produce from project to project may be similar in nature. The fact that more of the academic institutions' projects exceeded estimates by more than 20% may be the result of using inexperienced students to do much of the coding. Additionally, because most academic institutions are operated on a nonprofit basis, they may not be as concerned about making accurate estimates. One respondent wrote, "Most of our lessons are created by student-faculty teams with little restrictions. There is no internal accountability or contracting on a formal basis."

Estimation Methods. As shown in Table 13, 69% of private organizations distinguish between types of courseware when estimating CBT project costs while only 37% of academic institutions do (Chi square (1) = 10.16, $p < .05$). This may be seen as evidence of the greater need for private organizations to find ways to estimate accurately and yield a profit.

Table 13
Private and Academic Institutions that Distinguish Between Different Types of Courseware

		Distinguish between types	
Institution	n	Yes	No
Private	113	69%	31%
Academic	35	37%	63%

Measurement Unit. A chi square test showed significant differences (chi square (1) = 4.1, $p < .05$) between academic and private organizations and the units they use to measure courseware. Table 14 shows that private organizations tend to use a combination of ways to measure courseware more frequently than academic institutions. One possible explanation of this result is that private organizations' need to be profitable more than academic institutions, and therefore are using a combination of units to measure courseware in an attempt to find more accurate methods of estimating costs. In addition, academic institutions may find it easier to measure in hours because they almost always convert courses to CBT from classroom instruction.

Table 14
Private vs. Academic Institution and Unit of Courseware Measure

Institution	n	Single units of measure	Combination of measures
Private	113	55%	45%
Academic	35	74%	26%

Tasks Included. Table 15 shows that private organizations more frequently include "management time" as part of their cost estimates than do academic institutions (chi square (1) = 12.3, $p < .05$). This is probably the result of differences in the two organizations' cost accounting methods. Meetings and management time may not be seen as a cost to academic departments, but are significant cost factors within private organizations.

Table 15 also shows that private organizations more frequently include "learning the content" as part of their cost estimates than do academic institutions (chi square (1) = 4.8, $p < .05$). If universities use department faculty or students to write courseware for their own disciplines, the amount of time to become familiar could be negligible compared to private organizations who use the same writers for content in several areas.

Table 15
Private vs. Academic Institutions and Differences in Tasks Included in Estimates

		Task	
		Management time	
Institution	n	Included	Not included
Private	115	68%	32%
Academic	34	32%	68%
		Learning the content	
		Included	Not included
Private	115	72%	28%
Academic	34	50%	50%

Development Times. Private organizations that measure courseware in instructional hours produce simple simulations in less than half the time that academic institutions require. The average development time for a private organization to develop a simple simulation was 121 hours compared to 289 hours for academic institutions ($t = 2.06$, $df = 56$, $p < .05$). There were no differences between the times required for complex simulations, drill and practice, and tutorials.

Cost Factors. Private organizations rated both the effect of the "complexity of the interface" and "programmer's experience" lower than did academic institutions. Private organizations rated "complexity of the interface" 3.7 compared to a rating of 4.4 for academic institutions ($t = 2.73$, $df = 132$, $p < .05$). Universities may more frequently develop courseware which employs experimental computer interface equipment, making their projects more sensitive to interface complexity. Private organizations rated "programmer's experience" 3.8 compared to a rating of 4.2 for academic institutions ($t = 2.01$, $df = 147$, $p < .05$). Academic organizations often use relatively inexperienced student programmers who may be less efficient than experienced programmers who have developed labor saving programming techniques.

Accurate vs. Inaccurate Estimators

Tasks Included. A chi square test showed that accurate and inaccurate estimators include the same tasks in the development process. The difference in accuracy may come from the amounts of time and relative weights assigned for the tasks by more accurate estimators.

Development times. The reports of developers are inconsistent about whether the accurate estimators are also the most efficient developers. As shown in Table 16, accurate estimators reported development times for courseware units below the mean minimum more frequently than inaccurate estimators (chi square (1) = 3.8, $p < .05$). However, there was no significant difference when they reported times by the type of courseware. The informal nature of the survey may have caused this discrepancy.

Table 16
Cost Estimation Accuracy and Minimum Development Times

Accuracy	n	Minimum development times: Below mean	Above mean
Estimates < 10%	31	68%	32%
Estimates exceed cost by > 10%	24	38%	62%

Cost Factors. Chi Square test showed that the only cost factor rated significantly different by accurate and inaccurate estimators was "ease of authoring system or language" (chi square (3) = 8, $p < .05$). The results listed in Table 17 show that 52% of the more accurate estimators rated the factor a 4 or 5 compared to 68% of the less accurate estimators.

Table 17
Cost Accuracy and Ease of Authoring Language

Accuracy	n	Effect on cost: LOW 1-2	3	4	HIGH 5
Estimates < 10%	75	20%	28%	23%	29%
Estimates exceed cost by > 10%	78	18%	14%	42%	26%

Summary

This section of the report describes the design and results of a survey conducted to gather information from almost 200 CBT developers. The main purposes of this descriptive survey were to: (1) identify how CBT developers currently estimate costs, (2) collect data on average CBT development times, and (3) identify the factors CBT

developers report as most responsible for affecting costs.

Based on issues identified in the review of resources, three major questions were addressed about the current methods CBT developers use to estimate development costs. A questionnaire was designed to collect information about current practices, opinions about their efficacy, and recommendations for alternative methods. The three questions are listed below.

- What method do CBT developers currently use for estimating CBT development costs?
- How much time is actually required to produce computer-based training?
- Which factors which have the highest effect on development times?

The emphasis of the data analysis was on descriptive statistics. The data were also analyzed by comparing the following groups:

- Experienced vs. inexperienced developers
- Academic vs. private developers
- Accurate vs. inaccurate developers

Among the major findings of this survey are:

- Only 9% of private organizations reported estimating costs within 5%.
- Experienced developers do not report any lower development times than do inexperienced developers.
- Inexperienced developers are twice as likely as experienced developers to be off in estimates by 20% or more.
- For respondents using the instructional hour as the unit of measure, the mean minimum development time for one unit of courseware was 140 hours, the mean maximum was 316 hours.

- The range of development times for the instructional hour was one to 4,000 hours.

- Developers do not include the same development tasks in their estimates of a unit of courseware.

- The seven highest rated cost factors were: complexity of instructional design strategy, nature and complexity of content, client's demand for revisions, complexity of screen interactions, complexity and volume of graphics, conditional branching complexity, development team's experience producing CBT.

- The most frequently used single measure of courseware is the instructional hour, but only 1% continue to recommend its use.

The responses to this survey indicate that many developers are dissatisfied with current costing practices and are unable to make accurate cost estimates. In both Part 1 and Part 2, developers and purchasers called for a tool that would make the process easier and more accurate. The next part of this report includes a review of existing and developing costing tools in CBT and related areas.

PART THREE: COST ESTIMATION MODELS

It has been established that both purchasers and developers of CBT would benefit from a CBT costing model. In this part, known available CBT development costing tools were examined, as were software development tools, to determine their possible relevance to CBT costing. Courseware development shares similarities with the more mature field of software development. Additionally, a software package which is designed to provide such a model for CBT costing, The CBT Analyst, was validated informally using data from nine completed CBT projects. A description of the tool and results of the informal validation are contained in this report.

Review of Existing Cost Estimation Models

Software Metrics

Cost estimation problems common to both software and courseware include determining the unit of measure and identifying factors that affect development time. The similarity of the problems is illustrated by the following two quotations concerning software development. "Software cost estimating is, at best, an imprecise art. Estimates are generally clouded by questions of instruction counts and complexity factors, and universally not believed" (Bergland and Gordon, 1980, p. 13). "Software costing will never be an exact science. Too many variables--human, technical, environmental, and political-- can affect the ultimate cost of software" (Pressman, 1982, p.66). An approach called software metrics has been developed to provide a systematic means of measuring software.

Software and courseware are both developed according to specifications for functions they must perform. Bergland and Gordon (1980) compared the process of designing software with that of designing a house. The problems faced by software developers are made more difficult because without a comprehensive analysis, there are no detailed design parameters. Like architects, software developers typically must bid on contracts prior to knowing every design parameter. "Frequently, major systems must be defined and bid before detailed functional sequences are specified. Consequently, the software estimates for these systems reflect the lack of design definition necessary for accuracy" (p. 13). In both software and courseware development, the front-end analysis, from which the necessary parameters are derived, is typically conducted after the contract is underway. Making estimates when many variables are still undefined leads to a high probability of cost estimation errors in both courseware and software development.

Courseware productivity is typically measured using personhours of development per hour of delivered instruction. Software productivity is typically measured using lines of code (LOC) or delivered executable machine instructions (DEMIs). These metrics are controversial for similar reasons, notably, the inadequacy of a single unit of measure. "The simplest (and most controversial) measure of productivity is the number of validated source lines produced per personmonth" (Pressman, 1982, p. 67).

Bergland et. al. (1980) stated that estimating the cost of developing software is made difficult by the metric that is used—the number of machine instructions. This metric provides the single best correlation with development cost, but if one used only this measurement, estimation errors of nearly 1000:1 are possible. Consequently, qualitative factors such as complexity and experience must also be included in a software cost model. Bergland and Gordon note that, in contrast, estimates of hardware development costs can usually be made using a few simple metrics. Although it is necessary to identify the qualitative factors that affect software productivity, it "does not guarantee a credible estimate since both the relative value of a given factor and the sensitivity of a particular job to that factor must also be estimated" (p. 14).

To deal with the issues in measuring software and estimating software development costs, the field of software metrics has emerged. Software metrics encompasses the factors, measurement, and models which allow the development of quality software and the accurate estimation of the cost of producing that software (Conte, Dunsmore, and Shen, 1986, p. 3).

Key development factors used in software estimation models are similar to those believed to affect CBT estimates. Pressman (1982), TRW, and RCA each developed cost estimating models which included from five to seven categories of factors such as program size, program attributes, hardware attributes, project attributes and environmental attributes. (A complete list of these categories of factors for these three models is listed in Appendix D.) Many of the factors mirror those identified for CBT.

Walston and Felix (1977) are credited with one of the earliest attempts to study software cost data under partially controlled conditions. They began by identifying 68 factors that may have been responsible for variations in the amount of effort required to produce software. Using multilinear regression analysis, they identified 29 of the factors that were significantly correlated with productivity. Project leaders were then asked to rate the extent to which these factors applied to their projects. The mean impact on productivity of each variable was then computed. Theoretically, these means could be used to estimate the cost of developing new projects. The accuracy of this model was not validated by its authors on its own database, but a study of a subset of the database showed the model to be a poor

predictor of cost (Belady and Lehman, 1976). Conte et al. (1985) note that the poor results may be due to the wide variety in projects studied.

To date, a number of software development cost models have been validated. Cost models such as RCA's, TRW's and Walston and Felix's were validated by comparing predicted costs of a project with actual cost data at the completion of the project. One such model, the Intermediate (2) Constructive Cost Model (COCOMO) was found to give excellent results with its own database of 63 projects (Boehm, 1984). Despite its accuracy on its own database, the model is not universally accepted. Criticisms include the number of factors (15) that must be estimated and the fact that it has not been validated on a different database (Conte, et al., 1985). The field of software development has begun to develop metrics that may prove helpful in costing projects. Similar courseware metrics may be developed to systematize the costing of CBT development projects. Recommendations for using a method similar to the one described above to develop a CBT development cost model will be made later in this report.

Current CBT Development Cost Models

As mentioned in Part One of this report, Gery (1987) has developed "the groundwork for [a CBT time and cost estimating methodology]". Although this model came out too late to be included in the validation section, it shows promise. Gery's method requires the developer to examine 37 factors in four categories for the project. The judgments about the variables are plotted on a grid. The resulting matrix shows how the project ranks on the continuum of all factors. The developer is directed to synthesize the judgments and find where they group on the matrix of development hours provided. Ranges are 85-100, 150 to 300, and 300+ development hours per instructional hour. Although the top range of her scale seems low based on data presented earlier in this report, her method contains the most comprehensive list of development cost factors yet available. Gery (1987) accurately states "the main value of this estimating process lies in the certainty that you haven't left anything out" (p. 188). The model has not yet been validated.

Several projects are currently underway to develop CBT development cost models. Most of these projects seek to identify factors which affect development cost and use those factors in a checklist format to be considered when making cost estimates. G. Macomber (personal communication, May 1987) plans to set a range of acceptable dollar figures for each level of complexity per hour of instruction. C. Steier (personal communication, May 1987) is developing a database of the costs and associated cost factors to be used in estimating costs of level 3 interactive videodisc. The Naval Training Systems Center (NTSC) is developing a costing model which assigns an acceptable range of development times to each of the development tasks based on the range of complexity possible for each

task (W. Wright, personal communication, May 1987). NTSC plans to use the tool to assess proposals for firm fixed price contracts.

Summary

There are several similarities between software and CBT development cost estimation practices. Both fields have experienced difficulties in precisely estimating development effort. A front-end analysis is required to fully understand and accurately bid the specifications; without this analysis up-front, estimates are often inaccurate. The measurement units (LOCs or DEMIs for software and the instructional hour for courseware) are controversial. Many factors must be considered in estimating cost, particularly quality factors. The factors and weights of each factor need to be validated in order for them to be useful for predicting project costs. In software metrics, Walston and Felix's first attempt was not validated, but the Intermediate COCOMO (2) was found accurate with its own database. Though not universally accepted, software metrics may provide a model for CBT metrics which will systematize the measurement of CBT and estimation of CBT development costs.

A few attempts have been made to systematically develop costing models for CBT development costs. Gery's model includes the most comprehensive list of cost factors, but like the others being developed, it has not been validated. In the next section, one commercially available software package for estimating courseware development costs is evaluated with actual CBT projects to determine if it can be used to predict costs for particular projects.

The CBT Analyst: An Informal Validation

The CBT Analyst (Park Row Software, 1987) software program is the only known commercially available tool designed to provide CBT development cost estimates. Since there is a recognized need for a CBT costing tool and this particular tool appears to be unique, an informal validation of *The CBT Analyst* was made using data from eight completed CBT projects. The purpose of this validation was to determine the accuracy of the tool by comparing its estimates of cost for specific projects with actual costs. Estimates within 10% were considered accurate for this validation.

Description of the Tool

The CBT Analyst is a computer program consisting of five sections which are designed "to help training developers and managers make decisions about Computer Based Training." (Park Row Software, 1987) The sections are:

- Selecting Courses Appropriate for CBT

- Selecting the Appropriate CBT System for a Given Application

- Estimating CBT Development Costs

- Determining Cost/benefits of CBT for an Application

- Predicting the Success of a CBT Project

The section on "Estimating CBT Development Costs" contains the cost estimator examined in this report. The program is on a personal computer diskette and is accompanied by a folder of brief documentation.

The program is a set of rules based on CBT development cost factors such as type of instruction, author experience, degree of learner control, anticipated revisions, etc. Responses are assigned weights which are added to a base score of zero.

The total score is matched against thresholds for ranges of development hours per instructional hour. For example, a score of 26 is in the range of 200 to 400 hours per hour. The ranges in the tool are: under 100, 100-200 hours, 200-400 hours, and 500+ development hours per instructional hour. The tool presents the user with the estimated development hours per hour of instruction, as well as a list of the factors, including weights for each, which contributed to the score. The tool allows the user to modify or add to the rules, weights, or thresholds as desired.

Method

Subjects. For this independent validation of the tool, nine subjects were selected from interested contacts in previous portions of this research. All were CBT developers with three to 20 years experience who had previously developed at least one CBT project. The size of training production facilities ranged from eight to 350 employees. Five of the projects were for military customers, two were for commercial clients, and one was for an academic customer. Additional criteria included subjects' access to and willingness to share development data, and willingness to participate without monetary compensation. Selected projects had to have been completed within the last two years. The authors promised to share scores from the tool and summary results of the report with the subjects.

Procedure. All subjects were asked to participate in a telephone interview to validate a CBT development costing tool. They were told that they would be required to answer questions about a recently completed project and provide actual data on the number of development hours required and the number of student instructional hours developed. Subjects were sent a printed copy of the questions from

the "Estimating CBT Development Costs" portion of the software. Subjects were guaranteed company, project, and personal anonymity, and asked to be as candid as possible.

Prior to the research interview, the researchers reviewed the process with the subjects and defined and quantified unclear terminology in the tool. Although the tool did not provide these clarifications, the subjects were given this information so that the focus of the validation would be on the factors, weights and thresholds as opposed to the language used. An example of the type of clarification given follows. One question asks the user to rate the experience level of the CBT author. Allowed responses include "very experienced," "some experience," or "no experience." For this validation, these responses were quantified as 5 or more years, 1 to 5 years, and less than 1 year, respectively. All quantifications and definitions given to the subjects were derived from personal communication with the tool's author, G. Kearsley (May, 1987).

Subjects were also given guidance, based on an interview with G. Kearsley (personal communication, May 1987), on which tasks to include in their calculations of development hours. Specifically, subjects were asked to include labor hours for all individuals involved in technical and managerial tasks required to develop the CBT from analysis through evaluation. Specifically excluded were labor hours required for implementation, revisions and updates. Subjects were asked to exclude tasks which may have been required by the project but were not directly related to CBT development (e.g., research or major technical reports, system documentation, research on new technology).

Once the subjects had all relevant information, an appointment was made for a lengthy phone interview. During this interview, subjects provided:

- a project description,
- the number of instruction hours produced, and
- the total number of development hours.

Subjects were questioned to ensure that development hours were figured correctly and that all hours spent on the project (whether or not they were billed to the customer) were reported and that the correct tasks were included, according to the guidelines noted above.

Next, subjects answered each of the questions presented in the software (Version 2.0) for the selected project. In addition, subjects noted any discrepancies between responses they would have given prior to undertaking the project and responses they were giving from the perspective of project completion.

Responses from each subject were entered into the software to obtain development estimates. A programming error did exist in Version 2.1 of the software, however it did not affect any of the projects used.

Results

Table 18 shows the development estimates derived from the software, compared to actual development hours required for each project. For three projects, the estimate range given by the tool was within 10% of the actual number of development hours per instructional hour required. The other six estimates ranged from 86% too low to [illegible] too high.

Discussion

The purpose of this part of the research was to evaluate whether The CBT Analyst could be used to accurately estimate costs of actual CBT projects. Its predictions indicated that the tool does not yield more accurate results than those obtained by using the industry average of 200-400 development hours per instructional hour. Even using the closest end of the estimate range rather than the median, only three of eight project estimates were within 10% of actual development times. It should be noted that although not specifically stated in the tool, Kearsley (personal communication, May 1987) noted that the tool is only designed to give a general idea of the required development times for the project. Additionally, most subjects found the tool to be ambiguous even with additional clarification, and most judged that it could not be used to make estimates for a particular project.

Based on the research done for this report, the tool is not accurate in making estimates for particular projects because it yields such large estimate ranges as to be almost meaningless. The CBT Analyst does have several strengths. It provides "ballpark" estimates which may be useful for novices attempting to determine how much time might be required for types of CBT to be developed. It requires the developer to consider factors in a CBT project that may not have been considered previously. In addition, the format of multiple choice questions with explanations of answers made it relatively easy to use. Any part of the tool, including the rules, weights, and thresholds can be modified by the user, making it adaptable to specific needs.

Most subjects thought the tool was a step in the right direction; but only one stated that it could be used, as is, to estimate project costs. Some subjects felt the tool had limited value in assisting novice CBT developers to identify and address some of the key factors impacting CBT development costs. Some indicated that if trials such as the one conducted for this study could be used to modify the tool, it would be more useful. Examples of comments from subjects include: "not scientific, but based on judgment calls"; "with two or three

Table 18

Actual Development Hours Compared To CBT Analyst Estimates

Actual Development Per Hour	CBT ANALYST Estimate	Project Content	Type of Client	Total Development Hours	Years Development Experience	Instructional Hours Developed	Special Project Factors
1. 118	200-400	Foreign Language	Military	4,700	10+	40	Full language glossary 2 Track voice & video used
2. 231	100-200	Sales & Management	Commercial	4,851	10	22	-
3. 972	100-200	Technical	Military	972	5	1	Single prototype lesson
4. 266	200-400	Technical Safety	Military	3,940	5	14.8	-
5. 271	200-400	Equipment Maintenance	Military	54,150	3	200	System beta site
6. 211	200-400	Finanical Software	Commercial	3,800	10+	18	-
7. 294	500+	Technical Training	Military	11,766	3	40	Experimental technology, Client revisions very high
8. 384	100-200	Basic Computer Concepts	Academic	192	4	.5	Student developers used
9. 312	100-200	Language	Commercial	25,000	10+	80	-

trials, it could be calibrated to provide a baseline from which to start"; and "shows a naive perspective."

Many subjects identified factors and weights besides those in the tool which they felt should be considered in project cost estimates. Examples of their suggestions included:

- audience characteristics
- client's role in the development process
- stability of the decision-making environment
- content stability of new technology involved
- number of meaningful student interactions
- degree of help provided
- client's experience with CBT
- political factors

Several of these have been identified earlier in this report as factors affecting cost.

Factors related to political issues were mentioned by several subjects. For example, one subject noted that when a client with little CBT experience desires a high level of involvement in the development process, more development time typically is required than would be the case for a client with CBT experience who does not require a high level of involvement. Other political considerations were also mentioned as factors. For example, several levels of the management may insist on being involved in the review process and give comments after the courseware is developed, when changes are most time-consuming. The tool addresses these issues in a limited way with a question about whether or not a single person would review content. The assumption is that multiple reviewers increase development time, although one subject noted that, particularly on a large project, having only one reviewer can create a bottleneck in the review process.

Based on previous research described in this report and the subjects' comments for this validation, it appears that The CBT Analyst does not include all of the major cost factors. Although many of the commonly identified factors are included (for example, developer experience, complexity of instructional strategy), other important factors (for example, clarity of project specifications, nature and complexity of content) were not. In addition, some factors

are included in the tool which do not seem to be important. For example, several subjects noted that one factor in the tool, whether or not color graphics will be included, does not affect cost. (They indicated that the percentage and nature of the graphics does impact cost.)

Subjects also suggested that responses in the tool be more quantifiable, because the available responses prevented them from giving correct answers. For example, one question asks whether the CBT will be primarily tutorial, simulation, etc. One subject suggested that the question permit the percentage of each type.

This difficulty is supported by a validation that Kearsley did on the tool using five CBT experts (personal communication, June 1987). He reported that the experts were unable to reach a consensus on a single answer for each question for five sample project specifications. According to Kearsley, the experts were unable to reach a consensus because "each person was comparing the case to his/her own experience." Kearsley reported that the experts did arrive at approximately the same estimate ranges overall for each sample project. No validation was conducted using data from actual CBT projects.

Subjects also critiqued the weights assigned to responses. For example, the tool assigns equivalent weights to the choice of simulations, color graphics, and 20% or more revisions. One developer indicated, for instance, that color graphics should be weighted significantly lower, while revisions should be weighted higher.

Subjects also had a difficult time understanding certain questions in the tool, even with clarification from the researchers (based on personal communication with the tool's author). For example, in the question asking if this is a new or existing course, some subjects were confused about whether or not it would be considered an existing course if it existed in book form but had to be changed substantially before it was developed into courseware.

The developer of the tool appears to assume that all projects consist of approximately the same development tasks, with approximately the same percentage of time allotted to each task using the Instructional Systems Design approach. However, in actual practice, there are many variations. At the minimum the tool might ask the developer which development tasks required for the project.

In addition, the tool is not sensitive to the range of times that may be required for the same task on different projects. For example, Loven (1987) states that complete interactive video production costs can range from $28,000 to $375,000 per disc. As another example, to convert an existing lecture-based course to CBT, authors may have to attend the lecture training to get the content or may be able to get it from workbooks.

Users of this program should be aware that many projects will include costs not covered by this tool's estimates. In addition to the cost of development of CBT, tasks such as research on the best delivery system, major technical reports for government clients, and developing an accompanying workbook or documentation. Other expenses such as software licenses may or may not be incorporated into an organization's already existing overhead.

The tool does not adequately address the issue of courseware quality as a factor affecting cost. A generally accepted definition of courseware does not exist, however, the following elements must certainly be included in a definition:

- Students achieve objectives to the desired level.
- Students maintain a positive attitude toward the instruction.
- The courseware is objectively "good" (e.g. contains meaningful interactions, responds to each student input, etc.)

The CBT Analyst addresses quality only indirectly. For example, one question asks if the CBT will be used in-house or sold commercially. Additional points for a commercial product are based on the assumption that commercial courseware is subject to more rigorous quality control than in-house courseware (G. Kearsley, personal communication, May, 1987). A more direct question might be, "How much quality control is required for this project?". Other questions, such as the complexity of answer analysis or branching may address quality indirectly. A question about the level of student mastery, in conjunction with a question about student prerequisite skills and content complexity, might begin to address the number of hours required to produce courseware that meets mastery objectives.

Recommendations for Modification of the Tool

Several modifications could increase the tool's utility and accuracy. First, research should be conducted to determine the factors most highly correlated with CBT development costs. Suggestions for this type of a study are given in the Recommendations portion of this report. These factors should be used in tools such as this.

Second, weights for particular answers should be brought more in line with the actual relative importance of the factors. Subjects pointed out that equal weights were given to answers which are not equal in effect on cost in their experience. Currently answers are assigned weights of -10, -5, 0, 1, 2, 3, 5, and 10 points, with most answers having a weight of 0 or five. For example, the use of color graphics, development of simulations, and no CBT team experience all

receive a weight of 5. One subject suggested that points of 2, 6 and 10 would be more appropriate for these factors.

Third, the range of estimates should be narrowed and expanded upward. In particular, the ranges of 200-400 and 500+ hours per hour are very broad. As noted earlier, these ranges give developers a ballpark idea of development time, but do not give an estimate accurate within 10%. Ranges of no more than 50-75 hours would be more helpful, but again, some research into actual project costs based on factors would be required. As the tool is currently written, no prediction can fall between 401 and 500 hours per hour, and anything beyond 500 hours per hour is a shot in the dark.

Fourth, questions and responses should be more clearly defined and quantifiable. Currently terms such as "tutorial" and "authoring language" are not in the glossary, and new developers may not know the definitions. Even experienced developers have a hard time deciding whether their project authors have "some experience" or are "very experienced". These answers are easily quantifiable. More mixed answers should be permitted. For example, a user must currently decide whether the course to be developed is completely new or already in existence. Another more realistic answer might be "course developed in another medium" or "part of course must be developed from scratch". The tool should be field tested on potential users rather than experts to ensure that the language is understandable.

Fifth, the documentation or program should explain how to mesh development hour estimates into total project costs or at least caution the user that some projects may require tasks that are not included in the software's estimate. A section might be programmed in that suggests additional tasks that might need to be done.

Finally, programming bugs should be corrected. In version 2.0, a development error has resulted in some factors not being added into the total. In some cases this results incorrect estimates according to the tool's own rules. In both cases the average user is unlikely to analyze the tool as closely and will never know that the estimate was not correct. In version 2.1, all of the factors are totaled, but one of the weights is incorrect according to the rules the author described (G. Kearsley, personal communication, May, 1987).

Summary

Several research projects are now being conducted on cost estimating methods for CBT and the related field of software development. To date, some cost models have been developed for software development. Because of the similarities between these two fields, many of the procedures developed by the more mature field of software metrics appear to be applicable to CBT.

One commercially available CBT cost model, The CBT Analyst was informally validated using nine completed CBT contracts. In its current form, The CBT Analyst does not yield accurate estimates more often than the application of industry averages. Although the format is user-friendly, the program does not produce accurate results even by its own standards. Of even greater importance, even though the factors used were derived from expert opinion, they reflect neither a research base nor tests using actual projects. Whereas the tool may assist novices in identifying key factors to consider in the estimation process, it can not be used with confidence to make even general estimates for a particular project. In fact, the novice must be cautioned that there are costs in addition to those addressed in the tool which must be considered when estimating a complete CBT project. All users of the tool, particularly beginners, must be sure they interpret terms correctly.

As a result of this informal validation, the authors suggest refinements in the tool which incorporate modifications suggested in the recommendations above. Further research is required to identify and validate all the issues before a validated version of such a tool can be developed.

SUMMARY

As computer-based training matures and gains popularity as a training medium, purchasers and developers of CBT have expressed concern that current cost estimating methods for courseware development are glaringly inadequate. Accurate cost estimates are equally essential for both purchasers and developers of CBT. Purchasers must have reliable methods of estimating CBT costs so they have a realistic basis for deciding whether CBT is cost-effective for particular training applications. Key factors which contribute to the cost of CBT must be clearly defined and weighted for purchasers to evaluate proposals. CBT developers have also expressed an acute need for effective cost-estimating tools, since methods for accurately estimating CBT development costs are an essential prerequisite for successful project planning and management. This report was undertaken to investigate the current state of CBT development costing methods and make recommendations for improvements in the current process.

This report consisted of three parts, each designed to contribute to the attainment of the above goals. Part One consisted of a review of the issues involved in estimating CBT development costs. Sources of information for this section included available literature, research in progress, CBT professionals, and anecdotal information from seven completed CBT projects. Part Two consisted of a survey of almost 200 CBT developers to determine cost estimating practices, identify and rank factors impacting development costs, and collect data on average CBT development times. Part Three included a review of cost models for CBT and the related field of software development, as well as an informal validation of an existing CBT development costing tool, <u>The CBT Analyst</u>.

Data were collected from a variety of sources including professional literature, personal contacts with experts and practicing CBT developers and a survey of almost 200 developers. Much of the data, including the survey data, was reported by subjects from memory. Their responses provide insight into current practices and issues in costing CBT development but should not be considered as hard and cold facts.

Most developers report that they are unable to make accurate estimates for CBT development times. Less than 10% are able to estimate within 5% of actual cost. Experienced developers are no better overall at making accurate estimates, but inexperienced developers are more often off by over 20%. However, there is evidence that experienced estimators of development times are better at making estimates than those only experienced in CBT production. Because of poor estimates, many developers are only breaking even or actually losing money on CBT development contracts. De-scoping and requests for additional funds are the norm, and purchasers often do not receive what they expected. However, there is an indication that developers can improve estimates by considering project complexity factors when making estimates.

There are many reasons for inaccurate bids. There is no standardized estimation method and few historical records are readily available. Even when historical data do exist, developers may be reluctant to use it because the numbers are unacceptably high in light of "industry averages" and they believe that future projects would not have similar problems. In addition, developers are often surprised by factors not anticipated in the beginning, such as changes in scope by the client or hardware or software changes.

Developers generally bid on CBT development projects with very little knowledge of the project details. Developers often underbid projects because they fail to anticipate factors such as the required complexity, quantity of courseware, or long review processes. Some developers are now insisting on a complete analysis and initial design before the development project is bid to insure that most of the underlying issues are identified initially and taken into account when making a bid.

No standard method for measuring CBT is currently used. Although the "instructional hour" is most commonly used, it is widely disliked by developers as inaccurate and not reflecting complexity, quality, and other factors. Purchasers using this method have little assurance that an hour of instruction will be an hour that teaches the objective or even that it will actually take an hour. Although the purchaser or developer may have in mind a number of hours they would like the instruction to take, it is often impossible to predict delivery time in the computer based medium.

In addition, not all developers include the same tasks when costing a unit of CBT, so when purchasers are reviewing proposals, they cannot be sure that the proposals will produce the same courseware. The exclusion of certain tasks, such as analysis, formative evaluation, and management, is likely to have a major effect on the quality and effectiveness of the courseware. Since the effectiveness of courseware is seldom guaranteed, and the development tasks are inconsistent, the review process may be the client's only way to impact quality.

Most professionals who contributed to this research would welcome both a standard method for measuring CBT that includes quality factors and a method for making estimates of CBT development times. Most believed that developing measurement and estimation tools would be complex, but valuable.

Developers used several methods of estimating the cost of a unit of instruction. Most commonly, "industry averages" of 100 to 400 hours per hour of instruction are used. The reported range of hours required to develop a unit of CBT was from 1 to 4000 hours per instructional hour, with 153 and 316 as the mean low and high in the survey. This is close to the industry average of 100 to 400 hours per hour. These broad ranges are virtually useless for estimation without a systematic way of narrowing down the range. Developers simply must be able to estimate within 10% of cost if profit is an issue. A few developers use their own historical databases to predict costs for projects, and a few others are developing costing tools that include cost factors.

A host of factors were identified during the research as contributing to CBT development costs. The factors mentioned most often and rated as having the greatest cost effect are:

- Complexity of instructional design strategy,
- Clarity of project specifications at outset and adherence to them,
- Complexity of content,
- Number of revisions/unexpected client revisions,
- Complexity and number of features (e.g., graphics and helps) and,
- Experience of the development team, individually and as as a group.

(Appendix E contains a list of all factors identified in all parts of this research.) Although several attempts have been made to quantify these factors to yield a unit estimate, none has been proven any more accurate than applying the industry averages.

One commercially available tool for estimating CBT development costs, The CBT Analyst, was found to be useful for novices who need an estimate of the cost range to make a decision about whether to go ahead with CBT. However, like other guidelines currently available, the tool cannot be used with confidence to predict the cost of any particular project. The tool, like Gery's method, does point developers in the right direction by forcing them to focus on the factors that can increase costs.

The more mature field of software development offers promise for the future development of an accurate CBT development costing tool. Because of similarities between software and courseware development, it seems reasonable to apply software metrics techniques to courseware. However, research is necessary before tool development can begin. Recommendations for the steps in creating a CBT development estimation tool are described in the next section.

RECOMMENDATIONS

Most of the approximately 300 CBT professionals who participated in this research indicate that they would welcome a tool to standardize the measurement of courseware and estimation of development costs. However, before such a tool can be developed, a number of key elements must be resolved, including data collection procedures, methods of CBT measurement, and identification of key development factors. The recommendations are

listed below.

Further Research

Further research is needed to identify and validate:

- the most effective means of collecting and reporting data,
- the most effective means of measuring CBT,
- standard descriptors for CBT,
- factors and weights that affect CBT development costs, and
- a method for measuring CBT quality.

Development data from CBT projects should be collected in an accessible database to be used on bidding future projects of a similar nature. Project managers should note major factors affecting cost increases or decreases for future reference. A government requirement of a more detailed accounting of development labor would provide external motivation. This accounting might require that labor hours be reported for each lesson and task as opposed to labor category, as is usually required.

Research should be conducted to structure and validate a cost estimating tool. To develop an accurate cost estimating tool, factors believed to impact the cost of CBT development effort, including quality, must be identified. (Appendix E lists all factors mentioned in the course of this research.) Scales should be developed for each factor and be tested for reliability. An example of an objective scale for the factor of programmer experience would be 0-2 years, 3-5 years, and 6 or more years. Sample completed projects would then be selected and rated on each factor's scale.

A multilinear regression could be used to identify the factors that are significantly correlated with project development time. The factors would then be validated by testing them with the original sample, and finally with a new sample. Once a cost model has been generated, the rules and weights associated with could be embedded into a computer program.

Validated tools for software estimation have evolved over several years. The development of a comprehensive and standardized tool for estimating CBT development costs is also like to take several years to complete. In the meantime developers require methods for measuring CBT and estimating development times. For purchasers who must make changes to existing training or who are developing training from scratch, lesson or objectives with specifications for level of complexity of branching and graphics may yield the most helpful measurements. Using this method, for example, a purchaser/developer might specify: "Four simulations with linear branching; two types of help; with 50% video, 40% graphic and 10% all text pages; with a

meaningful interaction on 50% of the pages; to meet the objectives of enabling the student to identify the causes of four common faults of the stated system with 90% of the students achieving 90% accuracy."

Purchasers who are converting existing classroom training to CBT without significant modifications may find that the instructional hour can still serve as a helpful measurement. In this case the CBT hour would be equivalent to the instruction given in one hour's time in the classroom. However, developers and purchasers should be aware that those using the instructional hour have found it to be the source of many misunderstandings.

Until an accurate costing model has been validated, developers need a more narrow range for estimating an hour than the industry average. Those who do not have a historical data base may find it useful to use Gery's method of examining courseware factors and applying development ranges. Caution should be used at the higher ranges of development hours. While Gery lists 300+ per hour as the highest range, developers have reported development hours of 400 to 4000 hours per hour when certain factors are present.

New developers should remember that there are some tasks which must often be done on a CBT project that are not part of the price of all of the units of courseware. These tasks and other costs (noted in Part 3 of this paper) must be added to the estimate for the development of CBT to ensure an accurate estimate.

Bidding and Procurement Methods

The methods for bidding and procuring government and commercial CBT contracts should be studied to see how the methods might be modified to facilitate more accurate cost estimates. Such a study would investigate the feasibility and benefits to be gained by:

- making RFPs more descriptively detailed,
- completing the analysis phase of the ISD process before bidding on the last four ISD phases, or
- contracting with developers to develop a portion of courseware content before bidding the entire contract.

Unanticipated revisions and changes in scope were mentioned most frequently as the reasons for inaccurate bids. All contracts should clearly state the type and extent of revisions which are to be considered part of the development process, who will bear the cost of such revisions, and how necessary revisions not specified in the contract will be handled.

A clear written agreement between the developer and purchaser concerning what is desired in the end, how it is to be accomplished, how any deviations from the plan will be handled, and who will pay for the deviations will

improve project outcomes. Some of the issues which need to be clarified are:

- the percent of students that must reach a certain criteria,
- the objectives to be taught,
- the maximum and minimum number of hours that the courseware should fill (if this is important),
- the features desired,
- The steps in the review process, including the amount and type of revisions to be made within the contract, and
- who will pay for revisions beyond those agreed upon.

Completion of the analysis and even a prototype lesson before the entire project is bid will ensure that most issues are clear to both parties and that the information is available for use in pricing. In some cases the analysis may be completed by the purchaser, although the developer will generally need at least a complete review of the analysis, and possibly a partial analysis to gather information not provided by the purchaser. In other cases, the developer may perform a front-end analysis as a separate contract prior to the procurement of the development project.

Caution must be used in separating the front-end analysis from the development project. Separate contracts with the government for analysis and development could lead to a lengthy procurement process. Additionally, in cases where the purchaser provides an analysis to the developer, care must be taken to ensure that all of the information that a CBT developer might need is provided or can be gathered. Toward this end, a checklist of information to be provided in development RFPs would ensure that essential standard information be supplied for all development contracts or that a task within the contract allows for this information to be collected. A standard analysis for CBT development should include topics such as objectives and performance criteria, audience entry behaviors, and characteristics and suggested instructional strategies and features to be included.

The development of a prototype lesson to be used as a basis for bidding may result in more accurate results. Government contracts which have a fixed dollar amount assigned for development often only specify scope of project generally. The prototype(s) could be used to determine exactly how much of each type of courseware could be developed. Caution must be used when using prototypes as the basis for bids. First, prototypes invariably take more time than subsequent lessons, so adjustments must be made for accurate estimates to be made. In addition, many contracts require very different types of courseware to be developed; in these cases more than one prototype

may be required.

Emphasis on Student Achievement

In the flurry of discussion about measuring CBT and estimating development costs, the most important question is often ignored: Do the students learn? There are no generally accepted standards for courseware quality, but certainly a major component of any definition is the students' achievement of the objectives. The measures of courseware and tools to estimate development costs must be built around the issue of quality if we are to make realistic estimates and fairly compare bids. Two companies can bid for the same number of instructional hours with very different prices. The question that must be addressed is: Will the students learn equally well? To this end, courseware developers may find it an advantage to guarantee results rather than a specific number of "instructional hours." Many of the comments from developers indicated that failure to clearly specify student outcomes contributes to ineffective courseware or excessive revisions. The focus of the RFP should be on providing all information necessary to ensure that the description of the desired student outcomes are detailed and unambiguous.

The focus on student achievement has other advantages as well. Developers and purchasers would learn to distinguish between strategies and features that help the student and those which are merely fashionable. A focus on the objectives could also reduce the number of revisions, as only changes that assist the student in meeting the objective need be made.

This focus on student achievement rests on one very critical task--the ability of developers and purchasers to determine at the beginning of the project, exactly what outcome for the student is desired. After communicating with almost 300 CBT professionals, the researchers believe that clearly specifying student outcomes would make the single most important contribution to the ability of developers to estimate costs accurately and produce computer-based training that teaches.

Computer-based training development is a highly competitive field, with many factors complicating the development process. Currently many developers are finding it difficult to make a profit, partly due to their inability to make accurate estimates, and partly due to a misunderstanding by all involved of the complexity and opportunities for cost far above industry averages. Experience in the field, as in the field of software development, will undoubtedly improve developers' and purchasers' ability to work together to develop effective training. The recommendations in this report represent essential steps for defining and systematizing the complex process of estimating CBT development costs. When they are implemented, these recommendations could bring about improvements in the cost estimating process which will be a boon to CBT developers and purchasers alike.

REFERENCES

Avner, A. (1979). Production of computer-based instructional materials. Issues in Instructional Systems Development, In H. F. O'Neil, Jr. (Ed.), (pp. 133-180). New York: Academic Press.

Avner, A., Smith, S., & Tenczar, P. (1984). CBI authoring tools: Effects on productivity and quality. Journal of Computer-based Instruction. 11, 85-89.

Belady, L.A. & Lehman, M.M. (1976). A model of large program development. IBM Systems Journal, 15(3).

Bergland, G., & Gordon, R. (1980). Tutorial: Software design strategies (2nd ed.). Los Angel Institute of Electrical and Electronic Engineers Computer Society.

Boehm, B.W. (1984). Software engineering economics. IEEE Transactions on Software Engineering. 10(1), 4-21.

Brooks, F. P. (1982). The mythical man-month: Essays on software engineering. Reading, MA: Addison-Wesley.

Buck, J. A., & Gillespie, L. (1985). Computer-based training (CBT) starter kit. Reston, VA: Longman Crown.

Buckle, J. (1977). Managing software projects. New York: American Elsevier Inc.

Conte, S., Dunsmore, H., and Shen, V. (1986). Software engineering metrics and models. Menlo Park, CA: Benjamin/Cummings Publishing Co.

Dean, C., & Whitlock, Q. (1983). A handbook of computer based training. London: Kogan Page.

Fairweather, P., & O'Neal, A. (1984). The impact of advanced authoring systems on CAI productivity. Journal of Computer-based Instruction. 11, 90-94.

Gailey, F. (1973, August). An analysis of development: Use time ratios for a computer assisted instruction unit on basic houseshold electricity. Paper presented at Association for the Development of Computer-Based Instructional Systems, Ann Arbor, MI.

Gery, G. (1986). How long will it really take?: Estimating CBT development costs. Data Training. 5(8), 31-36.

Gery, G. (1987). Making CBT Happen. Boston: Weingarten Publications.

Grimes, G.M. (1975). Cost of Initial Development of PLATO Instruction in Veterinary Medicine (CERL Report X-43). Urbana, IL: University of Illinois.

Hillelsohn, M. J. (1984). Benchmarking authoring systems. Journal of Computer-based Instruction. 11, 95-97.

Kearsley, G. (1985). The CBT advisor: An expert system program for making decisions about CBT. Performance and Instruction. November, 24(9), 15-17.

Kearsley, G. (1983). Computer-based training. Reading, MA: Addison-Wesley.

Loven, J. (1987, June). How Much Does Interactive Video Cost?. Paper presented at the Conference of the Society for Applied Learning Technology, Stamford, CT.

Mikos, R., Sullivan, D.J., Hebein, J.M., & Casey, R.J. (1987). Estimating Training Effort/Costs: Revisited. Performance and Instruction. 26(5), 24-29.

O'Neil, H. F. (Ed.). (1979). Issues in instructional systems development. New York: Academic Press.

O'Neil, H. F. (Ed.). (1979). Procedures for instructional systems development. New York: Academic Press.

Orlansky, J., & String, J. (1979). Cost effectiveness of computer-based instruction in military training. (Contract No. DAHC15-73-C-0200). Arlington, VA: Institute for Defense Analysis (Defense Technical Information Center IDA Paper P-1375).

Park Row Software. (1986). The CBT Analyst [computer program] LaJolla, CA: Park Row Software Press Services.

Pressman, R. S. (1982). Software engineering: A practitioner's approach. New York: McGraw-Hill.

Stembler, W.A. & Pond, G. (1987). Costing drivers for interactive courseware. In R.S. Staley II (Ed.), Proceedings of the 1987 Conference on Technology in Training and Education (pp. 446-453). American Defense Preparedness Association.

Walston, C. and Felix, C. (1977). A method of programming measurement and estimation. IBM Systems Journal, 16(1), 54-73.

BIBLIOGRAPHY

Adams, A., & Rayhawk, M. (1986). (In Preparation) A review of cost and training effectiveness analysis cost models. Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Alessi, S., & Trollip, S. (1985). Computer based instruction: Methods and development. Englewood Cliffs, NJ: Prentice-Hall.

Axelrod, C. W. (1982). Computer productivity: A planning guide for cost effective management. New York: John Wiley and Sons.

Begg, J., & Bernstein, K. (1986, August). An analysis of cost of computer based training hardware and courseware development for the model training program for Reserve units. (ARI Research Note 86-87). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A172 572)

Branson, R. K., Kruger, B. P., & Farr, B. J. (1985, March). Job skills education program: Cost benefit tradeoff analysis. (ARI Research Note 85-42). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A162 896)

Brown, J. (1984). Trends in instructional technology. Syracuse, NY: Syracuse University. (ERIC Document Reproduction Service No. ED247926).

Bruce, P., & Pederson, S. (1982). The software development project. New York: John Wiley and Sons.

Bunderson, C. (1970). Justifying CAI in mainline instruction. (Technical Memo No. 4). Austin: University of Texas, Computer Assisted Instruction Laboratory.

Burright, B. K. (1987, March). Using the unified airman record and occupational surveys to estimate training benefits. In J. Orlansky (Chair), Workshop on Training Cost-Effectiveness Data and Methodologies. Workshop conducted at the meeting of the Department of Defense and the National Security Industrial Association, Orlando, FL.

Falleur, D. (1984, July). SuperPILOT: A comprehensive computer-assisted instruction programming language for the Apple II computer. Paper presented at the annual meeting of the American Society for Medical Technology and American Medical Technologists, Kansas City, MO.

Frank, W. (1983). A guide to software economics, strategy, and profitability. New York: John Wiley and Sons.

Goldberg, I., & Khatri, N. (In Preparation). A review of models of cost and training effectiveness analysis (CTEA). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Hall, K. (1982). Computer-based education: The best of ERIC June 1976-1982. New York, NY: ERIC Clearinghouse on Information Resource, Syracuse University. (ERIC Document Reproduction Service No. ED232615).

Hawkins, C. (1977). Computer based learning: A survey of the factors influencing its initiation and development. Utrecht, The Netherlands: Utrecht State University.

Head, G. E. (1985). Training cost analysis: A practical guide. Washington, D.C.: Marlin.

Himwich, H. A. (1977). A comparison of the TICCIT and PLATO systems in a military setting. Urbana-Champaign, IL: University of Illinois. (NTIS No. AD-A051 696).

Karsch, A. (1975). Computer program input instructions for cost performance forecasting model. (Cost Research Report No. 117-A). Wright-Patterson, OH: Wright Patterson AFB, Aeronautical Systems Division.

Kearsley, G. (1982). Costs, benefits, and productivity in training systems. Reading, MA: Addison-Wesley.

Kearsley, G. (1984). Instructional design and authoring software. Journal of Instructional Development, (7)3, 11-15.

Kearsley, G. (1985). Training for tomorrow: Distributed learning through computers and communications technology. Reading, MA: Addison-Wesley.

Knapp, M. I., & Orlansky, J. (1983). A cost element structure for defense training (Paper P-1709). Arlington, VA: Institute for Defense Analysis.

Lee, C., & Zemke, R. (1987). How long does it take?, Training, 24(6), 75-80.

Levin, H. (1984). Costs and cost effectiveness of computer-assisted instruction. Palo Alto, CA: Stanford University, Institute for Research on Educational Finance and Governance.

Lyman, E., & Postlewait, D. (1983). A chronological bibliography of PLATO articles. Urbana-Champaign, IL: University of Illinois, Computer-based Education Research Laboratory.

Mason, E. J., Smith, T. A., & Gohs, F. X. (1982) Models for estimating costs of computerized instruction. Juneau, AK: The Office of Educational Technology and Telecommunications, Alaska Department of Education.

Merrill, D. (1985). Where is the authoring in authoring systems? Journal of Computer-based Instruction. 12, 90-96.

Moore, G. (1987). How Costly are Computer-Based Instructional Systems? A look at two approaches. Canadian Journal of Educational Communication. 16(1), 33-43.

Ridge, W., & Johnson, L. (1973). How to control costs and improve profits through the effective management of computer software. Homewood, IL: Dow-Jones, Irwin, Inc.

Seidel, R., & Kopstein, F. (1970). Resource allocations to effect operationally useful CAI. Paper presented at conference on Applications of Computers to Training, National Security Industrial Association, Washington, D.C.

Seidel, R. J., and Wagner, H. (1977). Cost-effectiveness specification for computer-based training systems (Contract No. MDA903-76-C-0210). Arlington, VA: Defense Advanced Research Projects Agency. (NTIS No. AD-A081-051).

Shooman, M. (1983). Software engineering: Design, reliability, management. New York: McGraw-Hill.

Steinberg, E. (1977). Critical incidents in the evolution of PLATO projects. (MTC Report No. 12). Urbana-Champaign, IL: University of Illinois.

Technion International, Inc. (1986). Data collection system for estimating software development cost. Kirtland AFB: U.S. Air Force.

U.S. Civil Service Commission. (1971). Computer assisted instruction: A general discussion and case study. Washington, D.C.: Bureau of Training.

Vecchiotti, R. (1977). A survey and analysis of military computer-based training systems, Volume II: A descriptive and predictive model for evaluating instructional systems. Arlington, VA: Defense Analysis Research Project Agency.

Voeller, R. (1976, August). General purpose cost distribution model for computer assisted instruction. Paper presented at the Association for the Development of Computer-Based Instructional Systems, Minneapolis, MN.

Williamson, M. (1986). Project costing with COCOMO1. AI Expert, November, 52-57.

APPENDIX A
CBT DEVELOPERS AND EXPERTS CONTACTED FOR THIS RESEARCH

Among the military contacts who contributed to this research were personnel associated with:

- Army Research Institute for the Behavioral and Social Sciences (various units)
- Air Training Command at Randolph Air Force Base
- Army Training Support Center at Fort Eustis
- Department of Defense Training and Performance Data Center (formerly the Training Data and Analysis Center)
- Keesler Air Force Base
- Naval Personnel Research and Development Center
- Naval Training Systems Center

There were also a number of former military training personnel contacted who currently hold university and industrial training positions.

CBT developers and experts who gave advice and opinion for the research include:

- Allen Avner, University of Illinois at Urbana
- Richard Beger, Naval Training Systems Center
- Carl Behmer, McDonnel Douglas Corporation
- Alfred Bork, University of California, Irvine
- Al Boudreanx, Training and Performance Data Center
- Richard Braby, Eagle Technology
- Robert Branson, Florida State University
- Michael Bryant, Defense Training Data Analysis Center
- John Buck
- Jerome Cowley, Control Data Corporation
- Abbas Dauabi, Florida State University
- Walter Dick, Florida State University
- Marcia Drew, Quest Learning
- Irving Fink, TTGXZ, Keesler AFB
- Franz Fauley, Advanced Systems, Inc.
- Robert Forshay, Advanced Systems, Inc.
- Gloria Gery, Gery Associates
- Ralph Gilstrap, Wicat Systems, Inc.
- [illegible] Goldberg, University of the District of Columbia
- [illegible] Goldberg, U.S. Army Research Institute, Fort Eustis
- [illegible] Hanson, Convergent Systems, Inc.
- [illegible] Harper, Advanced Systems, Inc.
- [illegible], Creativision, Inc.

- James Hassett, Brattle Systems, Inc.
- Ruth Hawkins, McDonnell Douglas Corporation
- Jesse Heines, University of Lowell
- John Heindel, Boeing Computer Services
- Fred Hoffsteader, University of Delaware
- Julie Horine, Arthur Anderson Corporation
- Roger Hudson, Concourse Corporation
- James Hutton, J.R. Hutton and Associates
- Jack Johnson, Keesler AFB
- Wilson Judd, McDonnell Douglas Aerospace Corporation
- Greg Kearsley, Park Row Associates
- John Kessler, U.S. Army Research Institute
- David Kibbey, University of Illinois
- Donald Kristiansen, U.S. Army Research Institute, Fort Knox
- Wayne Knight, Naval Training Systems Center
- John LaBarber, Air Training Command, Randoiph AFB
- Janet Lamb, Army Training Support Center
- Jennifer Lippincott, Spectrum Corporation
- Thomas J. Livoti, Unisys Corporation
- Gary Macomber
- Kathy Mambretti, Icon Associates, Inc.
- Frank McGoogan, Florida State University
- William Montague, NPRDC
- Harold O'Neil, University of Southern California
- Jesse Orlansky, Institute for Defense Analysis
- Kathleen Riehle, Courseware, Inc.
- Wilhemina Savenye, University of Texas at Austin
- Tom Schaefgers, Courseware Applications, Inc.
- Ed Schorer, American Society for Training and Development
- Steve Seide, AT&T
- Robert Seidel, U. S. Army Research Institute
- William Shurmand, TCHTG/TTICOC, Keesler AFB
- Stanley Smith, University of Illinois-Urbana
- Al Smode, Naval Training Systems Center
- Dennis Sullivan, United Airlines Services Corporation
- Clara Steier, Eagle Technology
- Ronald Tarr, Training and Performance Data Center
- Richard Thoreson, U.S. Army Research Institute
- Christie Vitale, U.S. Air Force
- Kevin Wadsworth, Pinnacle Courseware, Inc.
- William Walsh, Eagle Technology
- Lois Wilson, Ford Aerospace and Communications Corporation
- Wallace Wulfeck, Navy Personnel Research and Development Center
- Wayne Wright, NTSC
- Robert Yeager, Intercom, Inc.
- Jim Young, Training and Performance Data Center

APPENDIX B

COMPUTER-BASED TRAINING DEVELOPMENT COST QUESTIONNAIRE

Please complete this questionnaire and return it to Scientific Systems Inc. by September 12, 1986. Use the enclosed self-addressed stamped envelope. If you wish to make additional comments, write them on the reverse side of this questionnaire.

ORGANIZATION BACKGROUND

1. Please check the category that best describes your organization.

   _______ private company   _______ government agency
   _______ academic institution   _______ other, specify ____________

2. How many employees and consultants at your organization work in computer-based training (CBT)? ____

3. How many years has your organization been in the CBT business? ____

4. What percentage of CBT work does your organization do in each of the categories below?

   _____ custom government contracting   _____ custom private contracting

   _____ courseware for the general market   _____ in house training

   _____ other, specify ____

5. Of the CBT courseware that you produce, what percentage is developed in the following content areas?

   ______ technical   _____ vocational   ______ academic
   ______ sales   _____ managerial   ______ medical
   ______ interpersonal skills   _____ other, specify ________________

6. Which of the following do you use to develop your courseware? If you use more than one, please indicate percentages.

   _____A general purpose programming language (e.g. Pascal, Basic). Specify which________________

   _____An authoring language which requires programming code (e.g. PLATO, TenCore). Specify which________________

   _____An authoring system which only requires entry of text and graphics, or is menu driven (e.g. Educator, SAM). Give example____

OVER PLEASE

COST ESTIMATION

7. When you estimate the cost of a CBT project, do you distinguish between different types of courseware?
   [ ] yes   [ ] no

8. If you answered "yes" to question 5, list the categories of courseware for which you separately estimate cost.

   ______________________   ______________________
   ______________________   ______________________
   ______________________   ______________________

9. What types of lesson models are typically produced by your organization? If you produce several types, indicate percentages.

   _____ drill and practice
   _____ tutorial
   _____ simple simulation (linear pathway)
   _____ complex simulation (complex branching)
   _____ other, specify ______________.
   _____ other, specify ______________.

> Definitions:
>
> Drills-cycle students through a series of problems, questions, definitions, etc.
>
> Tutorials-lead students through a socratic style of dialogue
>
> Simulations-place students in a controlled "real-life" situation in which they must bring the situation to some sort of resolution.

10. How accurate have you found you are in estimating the cost of CBT projects?
   [ ] within 5% of actual cost   [ ] within 10% of actual cost
   [ ] within 20% of actual cost   [ ] actual cost exceeds estimate by more than 20%

11. What have you found to be the major problem in accurately predicting CBT project costs?

12. What do you believe would be the ideal method of bidding CBT contracts?

13. When estimating costs for CBT projects, what <u>unit of measure</u> do you use?

[ ] number of expected student hours [ ] number of lessons
[ ] number of screens [ ] number of interactions per hour
[ ] other, specify ______________________________

14. What are the major advantages and disadvantages of the measure of CBT you use for estimating costs?

15. When you estimate the cost of a <u>unit</u> of CBT, which of the following are usually included in the estimate?

___ front end analysis	___ summative evaluation	___ formative evaluation
___ programming routines	___ writing lessons	___ computer down-time
___ programming lessons	___ learning content	___ meetings
___ video production	___ developing graphics	___ secretarial and
___ revisions	___ management time	other support
___ technical reports	___ computer operations	___ other ____________

16. Would you like to see an industry-wide standard system for measuring courseware? Why or why not?

17. If yes, what method of measurement would most accurately reflect true development times?

18. What is the range of hours that your organization actually requires to produce a unit of courseware?

OVER PLEASE

19. On the chart below, please estimate, based on your experience, the range of hours that each development activity in the courseware development process takes per unit. Complete columns for only the types of courseware you develop.

UNIT: (specify)________ ESTIMATED DEVELOPMENT HOURS / DEVELOPMENT ACTIVITIES	TYPE OF COURSEWARE: Drill & Practice	Tutorial	Simple Simulation	Complex Simulation	Other
1. Instructional Design (if separate from writing content)					
2. Writing content					
3. Writing programming code					
4. Creating graphics					
5. Storyboarding and producing video					
6. Reviewing and implementing in-house revisions					
7. Reviewing and implementing revisions requested by client					
8. Other					
9. Other					
10. Other					

VARIABLES AFFECTING COST

Below is a list of 29 variables which may affect the development of a unit of courseware. To the right of each variable, rate the effect the variable has on the cost of developing CBT by circling a number.

VARIABLES	MAJOR EFFECT ON COST <----			MINOR EFFECT ---->	
1. Nature and complexity of content	5	4	3	2	1
2. Type of behavior to be learned	5	4	3	2	1
3. Complexity of instructional design strategies (drill and practice, complex simulation, etc.)	5	4	3	2	1
4. Complexity of screen interactions	5	4	3	2	1
5. Conditional branching complexity	5	4	3	2	1
6. Response analysis complexity	5	4	3	2	1
7. Extent and complexity of feedback	5	4	3	2	1
8. Interfacing CBT with other systems (e.g., voice recognition)	5	4	3	2	1
9. Complexity and volume of graphics	5	4	3	2	1
10. Complexity and volume of video	5	4	3	2	1
11. Capabilities of authoring language or system	5	4	3	2	1
12. Ease of use of authoring language or system	5	4	3	2	1
13. Availability of reusable programming routines	5	4	3	2	1
14. Availability of text processor in authoring language or system	5	4	3	2	1
15. Author's (or SME) experience with content	5	4	3	2	1
16. Author's experience with instructional design process	5	4	3	2	1
17. Programmer's experience with programming language or system	5	4	3	2	1
18. Project manager's experience	5	4	3	2	1
19. Development team's experience producing CBT	5	4	3	2	1
20. Amount of client control over project	5	4	3	2	1
21. Client's demand for revisions	5	4	3	2	1
22. Turnover of CBT team and client	5	4	3	2	1
23. Number of people involved in the process	5	4	3	2	1
24. Client's available time and commitment to the project	5	4	3	2	1
25. Repetition of development tasks	5	4	3	2	1
26. Availability of in-house subject-matter expert	5	4	3	2	1
27. Interfacing CBT with other systems (eg. special keypad)	5	4	3	2	1
28. Other (specify) ____________ ____________	5	4	3	2	1
29. Other (specify) ____________ ____________	5	4	3	2	1

APPENDIX C

SOFTWARE DEVELOPMENT COST FACTORS

TRW lists 16 factors which are grouped into five categories. Some of these factors are:

1. program size - the number of delivered executable machine instructions
2. program attributes - complexity, type, language
3. hardware attributes - storage constraints
4. project attributes - personnel quality, experience
5. environmental attributes - requirements volatility, required quality

RCA developed a software cost estimating program which uses 42 factors grouped into 7 categories. The categories are:

1. project magnitude - amount of code to be produced
2. program application - type of project
3. level of new design and code - nct available from existing inventory
4. resources - experience and skill levels of developers
5. hardware limitations - memory constraints
6. customer specifications and reliability requirements - measures of reliability, testing and documentation required
7. development environment - what complicating factors exist

Pressman (1982) identified five categories of factors that influence the productivity of software. They include:

1. people factors - the size and expertise of the development organization

2. problem factors - the complexity of the problem to be solved and the number of changes in design constraints or requirements

3. process factors - analysis and design techniques used, languages available, and review procedures

4. product factors - reliability and performance of the computer system

5. resource factors - the availability of development tools, hardware and software resources

APPENDIX D:
ESTIMATING CBT DEVELOPMENT COSTS USING THE CBT ANALYST

Sample Project 5:
Equipment Maintenance Training (MTP-RC)

STEP 1: User answers questions for "Estimating CBT Development Costs" as presented in software. (Following are questions and answers for project 5).

Question 1: What type of CBT do you plan to develop? (If the course involves more than one type, indicate the primary type.)

	RESPONSE	RESPONSE DESCRIPTOR
	1: Tutorials	Tutorials (+1)
X	2: Simulations	Simulations (+5)
	3: Testing	Testing (+1)
	4: Embedded	Embedded (+5)
	5: Don´t know	Strategy Unknown

Question 2: How complex is the learning task the CBT course is to be developed for?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Complex learning task(s)	Complex Learning Task (+2)
X	2: Simple learning task(s)	Simple Learning Task
	3: Don´t know	Task Complexity Unknown

Question 3: Will color or graphics be used?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: Yes	Color/Graphics Involved (+5)
	2: No	No color/graphics
	3: Possible	Color/graphics possible

Note. This sample uses Version 2.1 (dated 6/25/87). Other projects used Version 2.0. Numbers indicate weights added or subtracted from score. Where no number appears, the answer does not result in a change on the score.

Question 4: Will interactive video or audio be used?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: Yes	Audio/Video involved (+5)
	2: No	Audio/Video not involved
	3: Possible	Audio/Video possible

Question 5: How will the courseware be developed?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Using a programming language	Programming language used (+5)
	2: Using an authoring language	Authoring language used (+3)
X	3: Using an authoring system	Authoring system used (+1)
	4: Don't know	Authoring method unknown

Question 6: Does a library of CBT routines and graphics exist or does all programming have to be done from scratch?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Yes	CBT library exists (-5)
X	2: No	No CBT library
	3: Don't know	CBT library unknown

Question 7: How much CBT experience does the designer (or design team) have?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Very experienced	CBT experience (+1)
X	2: Some experience	Some CBT experience (+3)
	3: No experience	No CBT experience (+5)

Question 8: How much experience does the developer/programmer have[a] with the authoring language or system being used?

[a] This question did not appear in Version 2.0 of the tool, so was not asked of most of the subjects during the interview. Subjects were contacted again or an answer was determined based on the subjects' project description.

RESPONSE		RESPONSE DESCRIPTOR
	1: Considerable	Authoring experience (-5)
	2: Some	Some authoring experience
X	3: None	No authoring experience
	4: Don't know	Authoring experience unknown

Question 9: Is this a new or existing course?

RESPONSE		RESPONSE DESCRIPTOR
X	1: New Course	New Course (+5)
	2: Existing course	Existing course
	3: Don't know	(No descriptor)

Question 10: Is the subject matter for the course available or is it in the process of being developed?

RESPONSE		RESPONSE DESCRIPTOR
X	1: Available	Content available (+5 only if new course)
	2: Being developed	Content being developed
	3: Don't know	Content status unknown

Question 11: Is the CBT course being developed for internal use or will it be sold commercially?

RESPONSE		RESPONSE DESCRIPTOR
X	1: Internal use only	Internal CBT course
	2: Commercial product	Commercial CBT course (+5)
	3: Both	Commercial CBT course (+5)
	4: Don't know	Commercial status unknown

Question 12: What kind of branching will the course involve?

RESPONSE		RESPONSE DESCRIPTOR
	1: Very complex	Complex branching (+5)
	2: Moderately complex	Moderate branching
	3: Simple linear	Simple branching
X	4: Don't know	Branching complexity unknown

Question 13: Will the answer analysis be simple or complex?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Complex	Complex answer analysis (+5)
X	2: Simple	Simple answer analysis
	3: Don´t know	Answer analysis unknown

Question 14: What kind of response feedback will the course involve?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Simple	Simple feedback
	2: Complex	Complex feedback (+5)
X	3: Mixed	Mixed complexity feedback
	4: Don´t know	Level of feedback unknown

Question 15: How much learner control will the program have?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: High degree	High learner control (+5)
	2: Moderate degree	Moderate learner control
	3: Low degree	Low learner control
	4: Don´t know	Learner control unknown

Question 16: What percentage of the course do you anticipate having to revised each year?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: Under 5% annually	Low revision (+1)
	2: 5-20% annually	Medium revision (+3)
	3: Over 20% annually	High revision (+5)
	4: Don´t know	Amount revision unknown

Question 17: Does a well defined storyboard exist for the CBT course to be developed?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Yes	Storyboard exists (-5)
X	2: No	No storyboard
	3: Partially	Partial storyboard
	4: Don´t know	Storyboard unknown

Question 18: If the CBT is to be developed by a team, does this team have previous experience developing CBT courses together?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Yes	Experienced CBT team
X	2: No	No Team experience (+5)
	3: Does Not Apply	(No descriptor)
	4: Don't know	Team experience unknown

Question 19: Do written standards, guidelines, or procedures exist for CBT development and are they followed?

	RESPONSE	RESPONSE DESCRIPTOR
	1: Yes	CBT guidelines used
X	2: No	CBT guidelines not used (+5)
	3: Partially	CBT guidelines may be used
	4: Don't know	Use of guidelines unknown

Question 20: Is the development effort being managed by an individual with past experience managing CBT projects?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: Yes	Experienced CBT manager
	2: No	Inexperienced CBT manager (+5)
	3: Don't know	Experience of manager unknown

Question 21: Is there a single individual responsible for approving the course and revisions to be made?

	RESPONSE	RESPONSE DESCRIPTOR
X	1: Yes	Single decision-maker
	2: No	Decision-maker not clear (+5)
	3: Don't know	Decision-maker unknown

Question 22: How would you describe the motivation level of the designer/developer(s)?

		RESPONSE	RESPONSE DESCRIPTOR
X	1:	Very enthusiastic	High motivation level (-10)
	2:	Interested	Moderate motivation level
	3:	Just a job	Low motivation level
	4:	Don't know	Motivation level unknown

STEP 2: Three composite rules are applied to answers and scores changed accordingly.

Rule 32: Inadequate CBT Specification. (+10)

("unknown" on questions 1, 13, 15)
Not applicable to Project 5.

Rule 33: Human Factors Unknown. (+10)

("unknown" on questions 20, 21, 22)
Not applicable to Project 5.

Rule 34: Experience Unknown. (-5)[a]

("unknown" on questions 7, 8)
Not applicable to Project 5.

[a] Software indicates -5, a reduction in score. This is a bug. It should be +5, because Kearsley notes that unknown experience increases development time.

STEP 3: CBT Analyst uses rules to calculate score.

	(CBT Analyst)	(Hand Calculation)
Motivation Level	-10	-10
Use of Guidelines	5	5
No CBT Team Experience	5	5
Hi Learner Control	5	5
Lo Revision	1	1
Content Available	-5	-5
New Course	5	5
Authoring Experience	5	5
Some CBT Experience	3	5
Authoring System	1	1
Audio/Video Involved	5	5
Color/Graphics Used	5	5
Simulation Desirable	5	5
TOTAL SCORE	29	29

STEP 4: Software compares score to thresholds to determine estimate.

Thresholds	Development Hours Per Hour
-9999 to 0	Under 100
1 to 20	100 - 200
21 to 50	200 - 400
51 to 9999	500+

STEP 5: Software provides user with estimate and list of factors used to determine estimate (as shown in Step 3).

Estimate for Project 5 (MTP-RC)

29 points = 200 - 400 hours per hour

APPENDIX E

CBT DEVELOPMENT COST FACTORS

The following list includes all factors mentioned in all parts of the research as effecting CBT development time. The list may be used as the basis for further research to develop a CBT development costing model.

Courseware Variables

- Nature and complexity of learning material
- Level of objectives
- Instructional design strategies
- Nature and frequency of interactivity
- Conditional branching
- Nature and depth of feedback
- Nature and depth of testing
- Nature, complexity and volume of graphics/animation
- Testing requirements
- Courseware specification standards: quality, specificity, stability
- Other media integration (type and complexity)
- Recordkeeping requirements
- Amount and type of learner control
- Amount and type of help
- Amount and type of video to be produced
- Number of design strategies throughout the course
- Stability of content

Technical Variables

- Authoring tools: capabilities and limitations, ease of use, editors available
- Productivity tools available: automated design tools, text processor interfaces, flowcharting software, software interfaces
- Multimedia interfaces
- Availability of graphics library which can most likely be used for this project
- Delivery hardware limitations and capabilities
- Presentation system cost
- Degree to which authoring language or system has built-in structures that allow the features desired for this project
- Availability of a software library or code templates which can most

likely be used for this project

Project Scope

- Adequacy of calendar time allotted for project
- Amount of courseware to be developed
- Number of people to develop courseware
- Development tasks to be included in project and size of each task
- Non-development tasks to be included in project

Client Characteristics

- Client experience with CBT
- Amount of client involvement in development project
- Number of clients involved in decision making/review
- Client's willingness/ability to provide resources, time necessary
- Availability of subject-matter experts willing and able to spend time on project
- Number of key players who support the development effort
- Type of organization: academic, government, private
- Expected turnover in client decision-makers, SMEs, reviewers
- Customer's predetermined ideas of what courseware should be
- Amount of experience working with this developer before
- Degree of customer satisfaction required
- Likelihood that client will understand and sign off on role definitions
- Degree to which client contact has power to get client reviewer, decision makers, SMEs to respond quickly and not change their minds
- Client contact's management skill
- The degree of likelihood that reviewers will stick with signoff on design and development specs
- Client commitment to project

Developer Characteristics

- Percent of development team that has worked together before on CBT projects
- Years of experience of manager
- Percent of design team with experience as CBT designers

- Percent of programmers/authors with experience using language or system being used for project
- Percent of writers/authors with subject-matter expertise
- Percent of writers/authors with good writing skills
- Percent of programmers with programming experience
- Percent of graphics designers with experience using graphics editor
- Percent of graphics designers with experience developing graphics for CBT
- Staffing approach: team with a variety of roles or one person does all tasks
- Corporate experience in the CBT business
- Type of organization: private, academic, government
- Expected turnover in development staff, managers
- Pressure from management to underbid project to get job
- Ease of flow of communication up and down
- Pressure to complete project without necessary information or resources
- Documented and used procedure and standards for review process that client has signed off on.
- Percent of development staff that must be trained using project funds
- Percentage of team members' time dedicated to this project
- Team synergy
- Total dollar amount of CBT projects on which developer has made a profit or delivered on time and in budget
- Structured design and development procedural understood and followed by team members on a previous project (for example, ISD)
- The degree of likelihood that reviewers will stick with signoff on design and development specs
- The number of reviewers, including managers, involved in the project

Quality Factors

- Percent of students that must achieve mastery
- Mastery level
- Level of bugs acceptable
- Percent of total screens with meaningful interactions
- Elements of good instruction that must be included (per Gagne)

Content/Audience Variables

- Audience homogeneity
- Audience familiarity with computers
- Stability of content
- Degree to which content must be created from scratch
- Percent of audience that have prerequisite skills